SELECTIVE EVOLUTIONARY GENERATION SYSTEMS: THEORY AND APPLICATIONS

by

Amor A. Menezes

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Aerospace Engineering)
in the University of Michigan
2010

Doctoral Committee:

Professor Pierre T. Kabamba, Chair
Professor N. Harris McClamroch
Professor Semyon M. Meerkov
Professor A. Galip Ulsoy
Associate Professor Ella M. Atkins
Assistant Professor Anouck R. Girard
Professor Gregory S. Chirikjian, The Johns Hopkins University

When searching for one that is a maximal,

Evolution is surely not cost optimal?

But rationality does show,

This is true for Monte Carlo,

And what's more, you get resilience as well!

To my parents,

whose great sacrifices opened doors of opportunity for me,

and to my sister,

whose efforts and accomplishments are a continual inspiration.

ACKNOWLEDGMENTS

This dissertation was made possible by the support of numerous people and organizations. I am indebted to Prof. Pierre Kabamba for the invaluable guidance and direction that he provided to me during the course of my Ph.D. study. I am also grateful to Prof. Semyon Meerkov for his thoughtful advice and insight. I thank the other members of my dissertation committee, Prof. Ella Atkins, Prof. Gregory Chirikjian, Prof. Anouck Girard, Prof. Harris McClamroch, and Prof. Galip Ulsoy, for their technical suggestions that improved the quality of this dissertation. I am also appreciative of the interactions that I had with other control systems faculty, including Prof. Dennis Bernstein, Prof. Domitilla Del Vecchio, Prof. James Freudenberg, Prof. Jessy Grizzle, Prof. Stéphane Lafortune, and Prof. Demosthenis Teneketzis, because these interactions helped shape this work.

I owe much of my understanding of flapping wing flight to the efforts of Dr. Hikaru Aono, Mr. Chang-Kwon Kang, Prof. Wei Shyy, and Mr. Patrick Trizila. I am particularly grateful to Mr. Kang and Mr. Trizila for the significant assistance that they gave me as I adapted their doctoral researches for my own purposes.

While researching and writing this dissertation, I benefited from the tremendous help and kindness of several staff members of the Department of Aerospace Engineering: Mr. Christopher Chartier, Mr. Thomas Griffin, Mr. David McLean, and Ms. Denise Phelps. These individuals went out of their way to smooth the path towards the completion of my degree, and I am exceedingly thankful for their actions. I would also like to thank Dr. Tershia Pinder-Grover and my fellow Graduate Student Instructor Mentors for the camaraderie that I experienced as a member of the Engineering

Graduate Student Mentor program.

The research in this dissertation was funded by various fellowships, assistantships and grants from the National Science Foundation, the Natural Sciences and Engineering Research Council of Canada, the University of Michigan Department of Aerospace Engineering, and the Xerox Corporation. The generosity of these organizations is sincerely appreciated.

I would also like to acknowledge the many well-wishes that I have received from friends and family all over the world. In fact, my Ph.D. study has given me two of my closest friends and cheerleaders in Dr. Eric Gustafson and Dr. Andrew Klesh. They shared the doctoral journey with me and made it a memorable experience too. Finally, I am ever grateful to my parents and sister for their encouragement and support throughout this worthwhile endeavor; they are the reason for my success, and this work is dedicated to them.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

FIGURE

LIST OF TABLES

TABLE

LIST OF APPENDICES

APPENDIX

ABSTRACT

This dissertation is devoted to the problem of behavior design, which is a generalization of the standard global optimization problem: instead of generating the optimizer, the generalization produces, on the space of candidate optimizers, a probability density function referred to as the behavior. The generalization depends on a parameter, the level of selectivity, such that as this parameter tends to infinity, the behavior becomes a delta function at the location of the global optimizer. The motivation for this generalization is that traditional off-line global optimization is non-resilient and non-opportunistic. That is, traditional global optimization is unresponsive to perturbations of the objective function. On-line optimization methods that are more resilient and opportunistic than their off-line counterparts typically consist of the computationally expensive sequential repetition of off-line techniques. A novel approach to inexpensive resilience and opportunism is to utilize the theory of Selective Evolutionary Generation Systems (SEGS), which sequentially and probabilistically selects a candidate optimizer based on the ratio of the fitness values of two candidates and the level of selectivity. Using time-homogeneous, irreducible, ergodic Markov chains to model a sequence of local, and hence inexpensive, dynamic transitions, this dissertation proves that such transitions result in behavior that is called rational; such behavior is desirable because it can lead to both efficient search for an optimizer as well as resilient and opportunistic behavior. The dissertation also identifies system-theoretic properties of the proposed scheme, including equilibria, their stability and their optimality. Moreover, this dissertation demonstrates that the canonical genetic algorithm with fitness proportional selection and the (1+1) evolutionary strategy are particular cases of the scheme.

Applications in three areas illustrate the versatility of the SEGS theory: flight mechanics, control of dynamic systems, and artificial intelligence. In the first application, this dissertation uses SEGS to evolve gaits for flapping the wings of a flying vehicle, in a way that is resilient with respect to changes in flight conditions. In the second application, this dissertation uses SEGS to evolve a disturbance rejection controller for a xerographic process, in a way that is resilient with respect to internal parameter variations. In the third application, this dissertation uses SEGS to evolve finite-state automata that infer a regular grammar, in a way that is resilient with respect to changes in the grammar. In this latter application, simulation results indicate that speciation occurs, with evolved finite-state automata exhibiting specific physiology and morphology that enable grammar inference.

The dissertation results touch upon several open problems in the fields of artificial life, complex systems, artificial intelligence, and robotics.

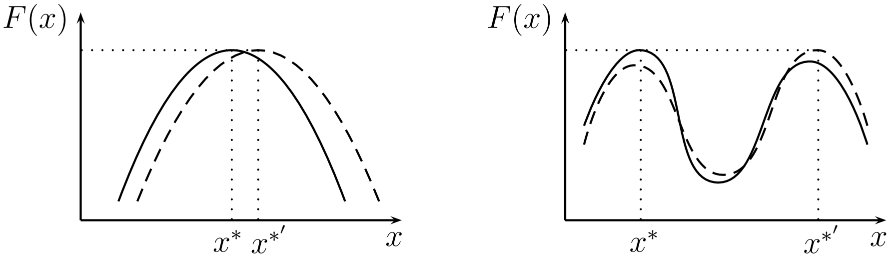
CHAPTER 1

INTRODUCTION

1.1 Motivation and Goals

This dissertation is devoted to the problem of efficiently designing an agent's behavior from a search space of possible actions such that the designed behavior is "good" and is also responsive to changes in what constitutes good behavior. Definitions of behavior design, efficiency, goodness and responsiveness are made in the following sections. An alternative formulation of this problem is the efficient determination of a viable species from a search space of possible organisms such that the species is fit and adapts to variations in its fitness landscape. A third version of this problem is the efficient optimization of an unknown objective function by finding a probability distribution on the search space of candidate optimizers (for instance, a delta function at the location that optimizes the objective function, i.e., *off-line optimization* [1]) using a scheme that can handle frequent perturbations of the objective function.

The second problem interpretation suggests evolution; however, 'Darwinian evolution may appear inefficient' [2]. The third problem interpretation suggests contemporary optimization schemes. Off-line optimization techniques [3–7], however, are non-responsive to perturbations of the objective function in two ways: *non-resilience* and *non-opportunism*, which are illustrated in Figures 1.1 (a) and (b). Specifically, small changes in the objective function may require changes in the probability distribution when the optimizer depends continuously or discontinuously on the perturbation, respectively. Hence, in practice, the objective function on which a candidate optimizer is implemented may be different from that for which the candidate optimizer was determined.



(a) Non-resilience (b) Non-opportunism

Figure 1.1: Off-line optimization strategies yield results that are non-resilient and non-opportunistic.

On-line optimization methods [8–14] are not currently designed for resilience and opportunism, but it is possible to develop on-line optimization methods that are more resilient and opportunistic than their off-line counterparts through the sequential repetition of off-line optimization techniques. However, such sequential repetitions are computationally expensive, a fact that may be shown by either an amortized analysis [15] or a competitive analysis [8, 9]. Moreover, these methods assume that the objective function is unvarying during a repetition, and they consequently incorporate observations about objective function changes, if any, between repetitions. Since the repetition schedule depends on the time each repetition takes to find a candidate optimizer, these methods are not truly on-line.

Alternative optimization approaches may involve reinforcement learning [16] and simulated annealing [17, 18], both of which are off-line and non-resilient. Techniques inspired by evolution (e.g., genetic algorithms [19–21], evolutionary strategies [22–25], and variations of these two) have the following features: 1) [26] notes that the use of 'the Darwinian principle does not guarantee successful optimization,' 2) [27] states that evolution optimization theories (on which these techniques are based) are an attempt to understand the diversity of life rather than demonstrate that organisms optimize, 3) [25] opines that 'searching for peaks depicts evolution as a slowly advancing, tedious, uncertain process,' and 4) evolutionary computation for dynamic fitness landscapes is a relatively new and uncharted area of study (for a recent overview, see [28]).

Therefore, the combination of efficient search and resilience appears difficult to achieve. Hence, the motivation for this dissertation is to design a behavior both efficiently and responsively. A behavior is sought that, when used in an evolutionary process under certain technical conditions, is a way of optimally searching for a desirable outcome, and is successful at finding a desirable outcome even if outcome desirability changes. Thus, the goals of this dissertation are:

- To present a behavior design technique that is efficient, on-line, inexpensive and yields behaviors that are resilient and opportunistic.
- To demonstrate the versatility of the technique through its application in problems from the robust control of dynamic systems, flight mechanics, and artificial intelligence.

1.2 Technical Approach and Impact

This dissertation demonstrates that *rational behavior* [29] is an example of the sought behavior. The work shows that the primary benefit of employing rational behavior is its capacity for optimal search, where optimality is defined as the quickest possible prior information trade off for reduced search effort. A secondary benefit is that rational behavior is a sufficient condition for resilience and opportunism. The work then proposes an on-line behavior design technique based on the novel concept of selective generation, which utilizes the ratio of the fitness values of two candidates and a parameter called the level of selectivity. In the limit as the level of selectivity tends to infinity, the scheme guarantees that the selected candidate is a global optimizer. Hence, the scheme presented in this dissertation is a generalization of standard optimization. Although rational behavior suggests dynamic transitions that are based on global knowledge, this dissertation proves that rationality may be achieved through a sequence of dynamic transitions using only local knowledge of the objective function. Thus, the proposed scheme is also computationally inexpensive at each step.

The implication is that the proposed scheme, when formulated appropriately, is capable of optimally searching for a desirable behavior, a fit species, or an optimizer of an objective function. Since the dissertation also shows that the proposed scheme generalizes genetic algorithms and evolutionary strategies, there is a potential broader impact in these fields as well. Moreover, rational behavior may also provide greater insight into natural evolution, with applicability to the evolution of artificial life.

1.3 Original Contributions

The original contributions of this work include the following.

- A novel mathematical definition of selection, the *Select* function, for use in behavior design.
- A demonstration that the canonical genetic algorithm with fitness proportional selection and the (1+1) evolutionary strategy are particular cases of a scheme utilizing the *Select* function.
- A proof that selective generation is a sufficient condition for rational behavior.
- A demonstration that rational behavior can lead to optimal search.
- A novel mathematical definition of resilience and opportunism.

- A proof that rational behavior is a sufficient condition for resilience and opportunism.
- An analysis of the effect that the level of selectivity has on resilience and opportunism.
- An examination of domains where resilience and opportunism are important.
- The identification of system-theoretic properties of a selective generation scheme, including equilibria and their stability and optimality properties.
- The evolution of flapping wing parameters for the purpose of hovering and trajectory-tracking by a flapping wing vehicle.
- A benchmark comparison of the canonical genetic algorithm with fitness proportional selection, the (1+1) evolutionary strategy, and the proposed selective generation scheme.
- The evolution of a controller that is robust to external disturbances or internal parameter variations.
- The evolution of finite-state automata that exhibit an elementary form of artificial intelligence: the inference of regular grammars.
- A demonstration that selective generation causes speciation among the grammar-inferring finite-state automata.
- A demonstration that selective generation results in the emergence of physiology and morphology among the grammar-inferring finite-state automata.
- A demonstration that selective generation ensures finite-state automata resilience and adaptability to changes in grammar.

1.4 Problem Definition

Let X be a search space. The problem of *behavior design* seeks 1) a probability density function (referred to as the behavior) $\phi_X : X \rightarrow \mathbb{R}^+$ that accomplishes specified objectives, and 2) dynamic transition laws that cause the variable x to be distributed according to ϕ_X, i.e., to exhibit the behavior specified by ϕ_X.

Let $z : X \rightarrow \mathbb{R}$ be an unknown, computable, and possibly changing function that we are interested in. Suppose that we are given an element Z in the image of z, and we wish to find $x \in X$ such that $z(x) = Z$, or such that $||z(x) - Z||$ is small. Formally, we want to design a behavior ϕ_X that achieves a known expected value Y, i.e.,

$$\mathrm{E}_{\phi_X}[||z(x) - Z||] = Y, \tag{1.1}$$

and we refer to this expectation as *goodness*. Let $y(x) = ||z(x) - Z||$.

We also desire the behavior ϕ_X to be responsive to perturbations in z, i.e.,

$$\frac{\partial \phi_X}{\partial z} \neq 0, \tag{1.2}$$

an inequality that will be later explored in more detail as *resilience and opportunism*.

We allow the behavior design method to employ a function $F : X \rightarrow \mathbb{R}^+$, a real-valued, positive fitness function that is a transformation of the function z. The scheme to find ϕ_X should be *efficient* in that it trades off prior information about X for search effort savings as quickly as possible. In the theoretical discussion that follows, we keep F arbitrary to maintain generality; however, we would also like to determine if efficient behavior design specifies the transformation between z and F.

1.5 Dissertation Outline

The remainder of the dissertation is as follows. Chapter 2 highlights the applicable literature. Chapter 3 presents the fundamentals of a theory of selective evolutionary generation systems that utilizes a novel scheme for fitness-based selection, and documents how selective evolutionary generation systems are different from other evolutionary computation strategies in the literature. Chapter 4 proves that a sufficient condition for resilience and opportunism is rational behavior, explains why rational behavior is desirable, demonstrates that resilience and opportunism may be achieved inexpensively at each step of the scheme, discusses the relationship with Markov Chain Monte Carlo methods, illustrates the theory by means of an analytically treated example, and extends the theory to facilitate a comparison with the literature. Chapter 5 utilizes the theory to optimize flapping wing gaits, and also benchmarks the proposed scheme. Chapter 6 applies the theory to evolve robust controllers for a dynamic system. Chapter 7 implements the theory to solve the problem of resilient grammar inference. Chapter 8 presents conclusions. Appendices A and B contain proofs of the theorems of Chapter 4 and selected pseudocode for Chapter 7, respectively.

CHAPTER 2

RELEVANT LITERATURE

The purpose of this chapter is to acquaint the reader with the current literature on the main topics of this dissertation. The following sections highlight notable references in a relevant field, and outline the relationship between the citations and this work. Due to the number of related fields and the existence of recent survey papers in each field, the sections in this chapter serve as selective guides to the literature rather than exhaustive descriptions. The applicable literature for secondary topics is deferred until these subjects are discussed.

2.1 Self-X Systems

Self-X systems are systems that are capable of self-assembly, self-organization, self-reconfiguration, self-repair, self-replication, or self-reproduction. The origins of this dissertation stem from the study of self-reproducing systems, a field inspired by the work of John von Neumann [30]. A comprehensive overview of self-replication is documented in [31–33]. In Aerospace Engineering, self-reproducing systems hold much promise for extraterrestrial colonization. Several space agency roadmaps, of which [34] is typical, suggest that individual countries will deploy advanced robots as-needed to expand the size of an established colony. However, it is well known that for every unit mass of payload to be launched into space, eighty additional units of mass are required to be launched as well [35] — hence, the motivation to endow robots with the capacity for self-reproduction. These machines would be able to utilize on-site resources to enlarge their numbers when deemed necessary for a given task. Extraterrestrial systems with such capability are less dependent than traditional colonies on the fiscal constraints of multiple launches of robots. Self-reproduction may therefore provide a highly cost-effective solution to the problem of establishing extraterrestrial colonies.

In a landmark conceptual study on a self-replicating lunar factory [36], a system that included paving, mining, casting, and mobile assembly and repair robots was proposed. Inspired by this work, [37] suggested a factory system comprising self-replicating multi-functional robots that could mine and transport materials and components within a lunar manufacturing facility. The work also demonstrated the feasibility of a self-replicating robot with a prototype made of LEGO Mindstorms components. At the same time (and in the years since), a number of researchers have developed modular self-replicating, self-assembling and/or self-reconfigurable robots (see, for instance, [38–51]). A current survey of the state of the art and the challenges facing modular, self-reconfigurable robot systems is given in the Grand Challenges of Robotics article [52], and in [53]. Other reviews are also available [54–56].

As the references above and those therein indicate, the focus has shifted to provable control of the modules of a single self-reconfigurable robot — the realization of various topologies [48], efficient and distributed control of a large number of modules [57, 58], recovery from module failures [59], and even module self-repair [60, 61]. Approaches for local control include reinforcement learning [62], cellular automata [63], and hormone-inspired swarming for self-organization [64]. This shift in focus to local control is due, in part, to the difficulty of achieving self-reproducing artificial systems in unstructured environments [65]. However, if one assumes that a colony of self-reproducing robots is technically feasible, then questions about the emergence of various behaviors arise (for example, [66] discusses the evolution of communication in robots).

By virtue of the harsh environment an extraterrestrial robotic colony operates in, self-reproducing robots need to learn, adapt, and possibly evolve to be tolerant of external disturbances that can affect the collective's overall goals. Hence, this dissertation examines the performance of self-reproducing entities that evolve both intelligence and resilience.

2.2 Resilience

The concept of resilience was first introduced in the seminal work [67], and a recent survey of the many definitions of resilience in the literature is available in [68]. Here, as in [69], two distinct meanings of resilience are identified: 1) *engineering resilience*, the time or rate at which a system returns to steady state equilibrium following a disturbance (i.e., a measure of disturbance recovery), and 2) *ecological resilience*, the amount of disturbance that can be absorbed by a system before transitioning to a new equilibrium (i.e., a measure of disturbance rejection). This dissertation examines both of these ideas, but does so after adopting a more general notion of resilience: a system is considered to be resilient if it exhibits a response to a disturbance. As long as such a response exists, the characteristic nature of the systems will ensure that either recovery from the disturbance takes place, or a transition to a new optimal equilibrium occurs.

Robustness in complex systems has been previously studied, using the Highly Optimized Tolerance conceptual framework for example [70, 71]. The results document complex systems that are generally robust, but an inescapable characteristic of the systems is their fragile nature, in that small disturbances can cause catastrophic cascading failures. However, there are numerous instances of autonomous robustness as well as resilience to small and large environment fluctuations in complex natural systems. Examples include physiological regulation in multi-cellular organisms [72, 73]; group regulation in colonies of social insects [74–76]; the evolution of species through adaptation and natural selection [77–79]; and the rebounding of complex systems from earthquakes, tsunamis, hurricanes, asteroid strikes, etc. [80, 81]. The apparent lack of resilience in a robust complex system further motivates this work.

2.3 Rational Behavior

The theory of rational behavior [82–84] deals with the axiomatic behavior of individual elements that take decisions in a decision space. The theory seeks to bypass complex calculations, favoring instead the computationally simplest mechanisms that result in rationality. Rationality here is defined as the selection of the most favorable decision among all possible decisions. For a recent comprehensive review of the field, see [85].

This dissertation utilizes the more general theory developed in [29]. A dynamic system with a decision space is *rational* if each trajectory of this system in the space is

1. *ergodic*: the trajectory explores all decisions in the decision space, and
2. *selective*: the trajectory slows down in the vicinity of the most advantageous decisions, i.e., the ratio of the mean time of stay of the trajectory in the vicinity of a more favorable decision to the mean time of stay of the trajectory in the vicinity of a less favorable decision is larger than unity.

Hence, the theory suggests the possibility of rapid convergence to the optimal state of a dynamic system. Unfortunately, global system knowledge may be required to determine how advantageous a state is.

The hypothesis that the theory of [29] yields additional benefits when suitably employed for optimization is validated in this dissertation. In addition, it is shown that local knowledge of the objective function is sufficient to guarantee rationality.

Reference [29] also sought to explain a remarkable property of the collectives that appeared in nature. These collectives, which had different fractions of professions (as in beehives for example), maintained an appropriate fractional distribution among the various social functions even if one of the castes was removed. Using fractional interactions, the theory examines the behavior of a collective and identifies the properties of systems of many elements. This is still an important topic; recent articles on the subject now incorporate the role of evolution and natural selection [86, 87]. This dissertation does indeed account for the processes of evolution and natural selection when demonstrating how to design a global behavior from local interactions.

It should be noted that the theory of rational behavior is not intended to model human rationality because ergodicity and selectivity may not define the behavior of an individual [88–92]. The term “rational behavior” arises in other contexts as well, for instance, game theory [93].

2.4 Evolution Theories

The re-discovery of Mendel’s genetic experiments with pea plants [94] led Fisher [95], Wright [96] and Haldane [97] to search for a mathematical model of Darwin’s theory of evolution and natural selection [77]. Their combined efforts laid the foundation for population genetics [98]. Notably, their work derives a formula for the probability that a mutation subject to natural selection becomes fixed in a population (see the summary [99]). They found that mutations that positively affect the number of offspring of a progenitor are selectively favored, mutations that negatively affect the number of offspring of a progenitor are selectively disfavored, and mutations that are neutral become fixed in a population with a probability equal to the initial frequency of the mutation. The prominent role offered to natural selection as a mechanism for evolution is reflected in the theory contained in this dissertation.

The neutral theory of molecular evolution (NTME) [100, 101] assumes that nearly all mutations are either highly deleterious or neutral. Since harmful mutations do not affect genetic variation within and between species, and adaptive mutations are rare and assumed to fix quickly in a population, it is the stochastic fixation of selectively neutral mutations that is the source of evolutionary change. NTME predicts that the rate of molecular evolution is independent of population size [99]. Although this dissertation does not consider neutral mutations to be responsible for the evolutionary process that is described, like NTME, population size is not considered to be a factor for evolution.

Neutral molecular evolution is also predicted by NTME to be linear in time, a useful fact for dating the relative divergence of different populations [99]. Similarly, this dissertation uses a system's generation index to directly measure time, a simplification that facilitates speciation analysis. In both NTME and this dissertation, evolution occurs more rapidly when selective pressures are low (i.e., the population fraction of neutral mutations in NTME is high) and more slowly when selective pressures are high (i.e., the population fraction of neutral mutations in NTME is low).

A relatively recent evolution theory defines quasispecies, the equilibrium mutant distribution that is generated by a specific mutation-selection process describing the erroneous replication of macromolecules [102, 103]. Template sequences for replication with different replication rates initially exist. The presence of mutations typically prevents the fastest replicating sequence from having the highest population frequency. Instead, erroneous replication can lead to the creation of new sequences or the duplication of other sequences in the ensemble. Unlike Darwinian evolution, selection operates on the whole quasispecies (the ensemble of mutants) rather than on an individual sequence [103]. Thus, fitness is a property of the quasispecies, and evolution is guided towards a known maximally fit sequence.

This dissertation produces a distribution of fit offspring too. However, the process acts in a way that is contrary to the hill-climbing quasispecies method described above, and does not make use of a known maximally fit element.

2.5 Grand Challenges and Open Problems

The original contributions in Section 1.3 — demonstrating the evolutionary emergence of intelligence, organization and resilience — touch upon a number of grand challenges and open problems documented in the literature. Below, we list several of these problems by quoting in italics from the sources cited, identifying the problem number in the associated reference, and outlining the relationship between the problem and this work.

1. Artificial Life [104]

   Problem 11: *Demonstrate the emergence of intelligence and mind in an artificial living system.* Here, the relationship is that grammar inference (Section 7.1) is an elementary form of artificial intelligence [2] and selective evolutionary generation mimics generation dynamics in living organisms [105].

2. Complex Systems [106]

   (a) Problem 2.1: *One of the most important principles is that global phenomena can emerge out of local interactions; how do we build artificial systems (or manage natural ones) so that the properties that emerge are the ones we want?* The selective evolutionary generation scheme employed in this work utilizes a sequence of local, inexpensive, dynamic transitions that result in global behaviors such as resilience and opportunism.

(b) Problem 2.4: *Optimization methods are usually preoccupied with finding the very best solution possible, whereas living organisms usually seek only "adequate" solutions. Genetic algorithms and other biologically inspired methods for search and optimization adopt a biological approach implicitly. However, the full requirements, and implications of, "adequacy" remain to be explored.* The notion of behavior design, which is addressed by the selective evolutionary generation technique proposed in this work, is a generalization of the standard global optimization problem: instead of generating the optimizer, the generalization produces, on the space of candidate optimizers, a probability density function referred to as the behavior. The generalization depends on a parameter, the level of selectivity, such that as this parameter tends to infinity, the behavior becomes a delta function at the location of the optimizer (see Chapter 4).

(c) Problem 2.5: *We suspect, but this has yet to be shown, that an evolutionary process occurs in which a process of random additions and collapses yields systems that are both complex and viable.* The discussion of this problem alludes to species diversity as a reflection of complexity. In Section 7.3.2, we experimentally show that the speciation of grammar-inferring finite-state automata occurs, and is a consequence of resilience to changes in the grammar.

(d) Problem 2.6: *A general question is whether specific conditions lead to particular kinds of structure (e.g. trees, cycles, etc.), organization and behavior.* The selectively evolved grammar-inferring finite-state automata all have common physiological and morphological traits that enable them to recognize the grammar correctly.

(e) Problem 3.3: *Disturbances such as fires and cometary impacts flip the landscape from a connected phase, in which selection predominates, to a* *disconnected phase, in which variation predominates. Many optimization algorithms (e.g., simulated annealing) exploit phase changes in the connectivity of the solution landscape to mediate between global and local search. Such analogies raise the prospect of identifying general processes that govern adaptive processes in many different kinds of systems.* The concept of a "phase change," either as a result of a disturbance or as a tool to transition between local and global searches, is also employed in selective evolutionary generation. The method results in phases where either selection or variation (i.e., ergodicity) is dominant; the changes between these phases suggest the phenomenon of *speciation*, defined later in this work.

3. Artificial Intelligence (AI) [107]
    (a) Rodney Brooks' Challenge 1: *Considerations of how future architectures might be designed so that software is self-configurable, and then even perhaps self-optimizing.* Selective evolutionary generation is inherently self-optimizing (Chapter 4). The results of this dissertation suggest that software that employs selective evolutionary generation of finite-state automata may lead to a solution to this challenge.
    (b) Tom Mitchell's Challenge 3: *Let's build agents that exhibit life-long machine learning, rather than machine learning algorithms that learn one thing and then get rebooted.* Since machine learning includes grammar inference [108, 109], this challenge requires that new grammars be recognized whenever they change, which is the focus of Chapter 7.
4. Robotics [52]

    Self-repairing subproblem of Grand Challenge 4 (modular self-reconfigurable robot systems): *Besides reconfiguring itself into a new shape, a system comprised of modular robots would be able to recover from serious damage, such* *as that which might result from an external collision or internal failure.* The sub-problem refers to two notions: that of resilience, which is at the heart of this work, and that of a diagnostic intelligence, which may use pattern recognition techniques to classify failure modes. Grammar-inferring finite-state automata algorithms were pioneered in the field of syntactic pattern recognition [110], validating the example in Chapter 7.

The preliminary results reported in this work suggest that a selective evolutionary strategy may be a tool for addressing the above open problems.

CHAPTER 3

THEORETICAL FOUNDATIONS OF SELECTIVE EVOLUTIONARY GENERATION SYSTEMS

3.1 Theory of Selective Evolutionary Generation Systems

The theory in this section is based on concepts from Generation Systems Theory (GST) [111]. GST formalizes the self-reproduction of *cells*, a term describing any entity that is capable of producing an offspring regardless of its physical nature. A robot, a bacterium, or even a piece of software code is considered to be a cell in this theory if they can each produce another robot, bacterium or some lines of code respectively. These cells utilize resources to self-reproduce. A selected resource is manipulated by the parent cell via an embedded generation action to produce an outcome.

We now extend these ideas to develop a theory of selective evolutionary generation systems. For behavior design, a cell is any element of the domain of the reward function and a resource is any input that facilitates a transition between cells. Furthermore, it is possible that resources are chosen probabilistically. Consistent with these notions, we make the following definition.

Definition 3.1. An *evolutionary generation system* is a quadruple $\mathcal{E} = (X, R, P, G)$, where

- X is a *set of n cells*, $X = \{x_1, x_2, \ldots, x_n\}$;
- R is a *set of m resources*, $R = \{r_1, r_2, \ldots, r_m\}$, that can be utilized for cell reproduction;
- $P : R \to (0, 1]$ is a *probability mass function on R*, given by $P(r_i) = \Pr[\mathcal{R} = r_i] = p_i$, $\sum\limits_{k=1}^{m} p_k = 1$; and
- $G : X \times R \to X$ is a *generation function* that maps a parent cell and a resource into a descendant cell outcome.

Use of the adjective *evolutionary* here is consistent with biology [105], where evolution is defined as the genetic changes in a biological population that occur every generation due to genetic changes from parent to descendant. Note that for each resource $r \in R$, we assume that an inexhaustible supply is available. This assumption is consistent with GST, which does not specify quantities of resources or cells.

Example 3.1. A random walk over $\mathbb{Z}^\nu$ is an example of an evolutionary generation system. Take $X = \mathbb{Z}^\nu$, $R = \{\pm\mathbf{e}_i,\ 1 \leq i \leq \nu\}$ (where $\mathbf{e}_i$ are the standard basis vectors for $\mathbb{Z}^\nu$), let P be the uniform probability distribution over R, and define $G : X \times R \to X : (x, r) \mapsto y = x + r$. The sequence of cells over the generations of this evolutionary generation system becomes a random walk.

Let $(r_\mu) = (r_1, r_2, \ldots, r_\mu)$ be a sequence of μ resources from R. We define the notation

$$G\left(x, (r_\mu)\right) := G(\ldots G(G(x, r_1), r_2) \ldots, r_\mu) \tag{3.1}$$

to denote the cell produced by x using sequence (r_μ). This is illustrated in Figure 3.1 as a directed graph.

Definition 3.2. The set of cells, X, of the evolutionary generation system $\mathcal{E} = (X, R, P, G)$ is *reachable* through G and R if, for all pairs $(x_1, x_2) \in X^2$, there exists $k \in \mathbb{N}$ and a sequence $(r_k) \in R$ such that $x_2 = G\left(x_1, (r_k)\right)$.

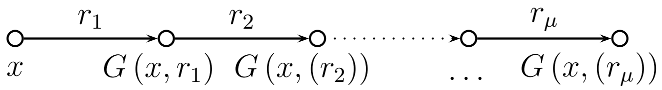


Figure 3.1: The directed graph of $G(x, (r_\mu))$.

Note that reachability of the cells of an evolutionary generation system is identical to that of reachability of the vertices of a directed graph in Graph Theory [112].

In Definition 3.1, the restriction that the offspring of a cell be itself a cell implies that the set of cells is *closed* [113], since there is no feasible transition to any element outside X. If the set of cells is also reachable, then X is said to be *irreducible* [113]. The previous example of a random walk over $\mathbb{Z}^\nu$ is an example of an irreducible evolutionary generation system.

We associate each cell with a non-zero, positive performance index that is a measure of the fitness of the cell, $F : X \to \mathbb{R}^+$. The notion of fitness facilitates the following novel mathematical definition of selection.

Definition 3.3. Given a cell set, X, and a fitness function $F : X \to \mathbb{R}^+$, let *Select* : $X \times X \times \mathbb{N} \to X$ be a random function such that if $x_1 \in X$ and $x_2 \in X$ are any two cells, and $N \in \mathbb{N}$ is the *level of selectivity*, then

$$\mathit{Select}(x_1, x_2, N) = \begin{cases} x_1 & \text{with probability } \frac{F(x_1)^N}{F(x_1)^N + F(x_2)^N}, \\ x_2 & \text{with probability } \frac{F(x_2)^N}{F(x_1)^N + F(x_2)^N}. \end{cases} \tag{3.2}$$

We can now define a selective evolutionary generation system (SEGS).

Definition 3.4. A *selective evolutionary generation system* is a quintuple $\Gamma = (X, R, P, G, F)$, where

- (X, R, P, G) is an evolutionary generation system;
- $F : X \to \mathbb{R}^+$ is a function that evaluates cell fitness;
- the set of cells, X, is reachable through G and R; and

- the dynamics of the system are given by

$$\mathcal{X}(t+1) = Select(\mathcal{X}(t), G(\mathcal{X}(t), \mathcal{R}(t)), N). \tag{3.3}$$

In (3.3), $\mathcal{X}(t)$ denotes the realization of a random cell variable at time t, $\mathcal{R}(t)$ denotes the realization of a random resource variable at time t, $G(\mathcal{X}(t), \mathcal{R}(t))$ denotes the offspring of the realized random cell utilizing the realized random resource at time t, and $\mathcal{X}(0)$ has a known probability mass function.

Also in (3.3), the probability of a cell realization at some future time given the present cell realization is conditionally independent of the past time history of cell realizations. Thus, the dynamics of a SEGS form a discrete-time homogeneous Markov chain [114]. This property is useful for the SEGS analysis conducted in Section 4.2.

The two central tenets of Darwin's theory of evolution [105] are embodied in Definition 3.4.

1) *Undirected variation* via the generation function. Permissible undirected variations include

   - *mutations* of all or part of a cell,
   - *recombination* of the constituent elements of a cell with the constituent elements of another (resource) cell,
   - *inheritance* of all or part of a cell when the generation function maps all or part of a cell to itself, and
   - *drift* of the constituent elements of a cell, as certain elements drift or become fixed due to the nature of the probability mass function over the resource set.

   That is, we impose no restrictions on the nature of the undirected variation process, thereby capturing all biological and computational mechanisms for creating diverse offspring. As we shall see, even *flow*, the sudden addition or removal of cells, is captured by our theory, since this process may be modeled by unexpected perturbations of the fitness function.

2) *Natural selection* via the *Select* function.

The *Select* function has a number of interesting properties, including:

- For all N,
$$\frac{\Pr[\mathit{Select}(x_1, x_2, N) = x_1]}{\Pr[\mathit{Select}(x_1, x_2, N) = x_2]} = \left(\frac{F(x_1)}{F(x_2)}\right)^N . \tag{3.4}$$
  That is, the ratio of the probabilities of selecting any two cells is equal to the ratio of their respective fitnesses raised to the power N. This property is called *local rationality.*

- For $N = 0$, the values of $F(x_1)$ and $F(x_2)$ are irrelevant. That is,
$$\Pr[\mathit{Select}(x_1, x_2, 0) = x_1] = 1/2, \text{ and} \tag{3.5}$$
$$\Pr[\mathit{Select}(x_1, x_2, 0) = x_2] = 1/2. \tag{3.6}$$

- When $N \to \infty$, if $F(x_1) > F(x_2)$ then
$$\Pr[\mathit{Select}(x_1, x_2, N) = x_1] \to 1. \tag{3.7}$$
  On the other hand, if $F(x_1) < F(x_2)$ then
$$\Pr[\mathit{Select}(x_1, x_2, N) = x_2] \to 1. \tag{3.8}$$

- If $F(x_1) = F(x_2)$ then, for all N,
$$\Pr[\mathit{Select}(x_1, x_2, N) = x_1] = 1/2, \text{ and} \tag{3.9}$$
$$\Pr[\mathit{Select}(x_1, x_2, N) = x_2] = 1/2. \tag{3.10}$$

The level of selectivity, N, has a biological interpretation as well. Suppose that the fitness of a cell is measured by the total number of descendants produced over k generations, $k \geq 1$. This prolificity is typically called *future reproductive value* or *fecundity* [105]. When a colony is initiated by two self-reproducing progenitors x_1 and x_2, the ratio of the descendant population fractions after k generations equals the ratio of the respective future reproductive values,

$$\left(\frac{F(x_1)}{F(x_2)}\right). \tag{3.11}$$

After k generations, the ratio of the probability of choosing, by random sampling, a descendant of x_1 to the probability of choosing a descendant of x_2 is equal to the ratio of the descendant population fractions (3.11). Correspondingly, the ratio of the probability of selecting x_1 at the initial time to the probability of selecting x_2 at the initial time, (3.4), is identical to the ratio of the respective prolificities, (3.11), with $N = 1$.

Now consider the following sequence of operations.

1. Initiate a colony with two self-reproducing progenitors x_1 and x_2, and let descendants be produced for k generations.
2. Extract a sample from the resulting population. Use the sample to initiate a second colony, and let descendants be produced for k generations.
3. Iterate the sample and colony initiation procedure until an N^{th} colony is produced.

Then, the ratio of the probability of selecting a descendant of x_1 to the probability of selecting a descendant of x_2 using this multi-step process becomes

$$\left(\frac{F(x_1)}{F(x_2)}\right)\left(\frac{F(x_1)}{F(x_2)}\right)\cdots\left(\frac{F(x_1)}{F(x_2)}\right) = \left(\frac{F(x_1)}{F(x_2)}\right)^N, \tag{3.12}$$

and it is now clear that N represents the number of selections that are made, assuming a k-generation fecundity interpretation of fitness.

A recent, well-publicized, biological experiment that fits this multi-selection model is [115]. Two polyethylene degrading strains of bacteria were isolated in this study as a result of the repeated selections of the progeny of soil bacteria that were forced to feed on a polyethylene enriched medium. The biodegradation of plastic is an example of an important engineering problem that, while previously difficult to solve, can now be tackled with the novel approach in this dissertation.

3.2 Comparative Literature Study

A SEGS as described by Definition 3.4 can be utilized as an evolutionary optimization algorithm (Section 4.2) to take advantage of its guaranteed properties. There are works in the optimization literature that appear to be similar: reinforcement learning [16], simulated annealing [17, 18], genetic algorithms [19–21], and evolutionary strategies [22–25]. Comparisons between these optimization methodologies and a SEGS approach can be made, and this section is devoted to providing such comparisons to outline the distinctions between approaches. For each of the optimization methodologies, we quantify the ratio of the probability of selecting a candidate optimizer of the objective function to the probability of selecting the optimizer's offspring. By comparing this resultant ratio to (3.4), we demonstrate the originality of our theory of evolutionary generation systems.

3.2.1 Reinforcement Learning

In reinforcement learning (RL) [16], a decision-making agent takes actions in an environment and receives a corresponding reward. The traditional RL problem is to determine the best policy or sequence of actions that maximizes the total reward.

There are two major differences between our work and RL. First, evolutionary generation systems theory does not allow changes in the tactics of individual cells, since there are fixed, probabilistic rules for generation outcomes in place. That is, the generation action taken by a particular cell is always the same but the outcome varies probabilistically due to selection. Second, RL seeks the long-term maximization of reward of a policy of cell-action pairs, while a SEGS focuses on probabilistically increasing the short-term reward from one cell-action pair. The different goals have consequences for responsiveness: an RL approach may not adapt the optimal policy if individual cell-action pair rewards are perturbed.

To facilitate a comparison between RL and a SEGS, consider the following deterministic reinforcement learning problem. Let x_1 and x_2 be the labels of two terminal cells, and let the current cell, also labeled x_1, be capable of a one-step transition to either of the two terminal cells. Hence, there are two possible policies: 1) a transition from x_1 to x_1, and 2) a transition from x_1 to x_2. Let the reward of cell x_i be $F(x_i)$. Using value iteration, the cost-to-go of the current cell with policy 1) is

$$V_1 = F(x_1) + F(x_1) = 2F(x_1), \tag{3.13}$$

and the cost-to-go of the current cell with policy 2) is

$$V_2 = F(x_1) + F(x_2). \tag{3.14}$$

Since RL chooses the policy with maximum reward, the ratio of the probability of selecting the terminal cell x_1 to the probability of selecting the terminal cell x_2 is

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{\operatorname{ind}\left(2F\left(x_1\right) > F\left(x_1\right) + F\left(x_2\right)\right)}{\operatorname{ind}\left(F\left(x_1\right) + F\left(x_2\right) \geq 2F\left(x_1\right)\right)}, \tag{3.15}$$

where ind denotes the indicator function, satisfying

$$\operatorname{ind}(True) = 1, \tag{3.16}$$

$$\operatorname{ind}(False) = 0. \tag{3.17}$$

The ratio in (3.15) is taken to be ∞ if the denominator is zero. This ratio is different from (3.4).

3.2.2 Simulated Annealing

The simulated annealing algorithm [18] randomly samples the search space at x_i, evaluates $F(x_i)$, and accepts new candidate optimizers x_i according to the Metropolis criterion. This criterion specifies that cells with better fitness are always accepted, while less fit cells are accepted with a probability that depends on the relative fitness with respect to the current cell x_1, and a "temperature" parameter T. The equation for the probability of selecting the less fit cell, x_2, is

$$\Pr[x_2 \text{ is selected}] = p_{SA} = \exp\left(\frac{F\left(x_2\right) - F\left(x_1\right)}{T}\right), \tag{3.18}$$

whenever $F\left(x_1\right) > F\left(x_2\right)$. Therefore, decreasing the temperature or increasing the relative fitness decreases the acceptance probability of less fit states.

The ratio of the probability of selecting the current candidate optimizer x_1 to the probability of selecting another candidate optimizer x_2 is

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{1 - \left(\operatorname{ind}\left(F\left(x_2\right) \geq F\left(x_1\right)\right) + \operatorname{ind}\left(F\left(x_1\right) > F\left(x_2\right)\right) p_{SA}\right)}{\operatorname{ind}\left(F\left(x_2\right) \geq F\left(x_1\right)\right) + \operatorname{ind}\left(F\left(x_1\right) > F\left(x_2\right)\right) p_{SA}}. \tag{3.19}$$

This ratio is different from (3.4).

3.2.3 Genetic Algorithms

The canonical genetic algorithm [19] models each cell of the search space, x_i, as a binary string of length l to which a fitness value $F(x_i)$ is associated. The algorithm outline [116] follows:

```
1: choose an initial population
2: determine the fitness of each individual
3: perform selection
4: repeat
5:     perform crossover
6:     perform mutation
7:     determine the fitness of each individual
8:     perform selection
9: until some stopping criterion applies
```

We are interested in the probability that a cell, x_1, of the population at Line 4 is chosen to be a member of the population for the next generation (i.e., after one iteration of the repeat loop) without experiencing crossover or mutation. We then compare this probability to the probability that an offspring of x_1 is a member of the population at the next generation. Let the probability of crossover of x_1 with another binary string be $p_c \in (0, 1)$, and let mutation of the j-th bit of x_1 occur independently with probability $p_m \in (0, 1)$.

Of the many kinds of selection processes (e.g., fitness-proportional selection, tournament selection, or truncation selection) that can be applied to the cells of a population, Y, let us first consider fitness-proportional selection. The probability of considering x_1 with this selection process is

$$\Pr[x_1 \text{ is considered}] = \frac{F(x_1)}{\sum\limits_{y \in Y} F(y)}. \tag{3.20}$$

Hence, the probability that an unchanged candidate optimizer, x_1, is a member of the population for the next generation is

$$\Pr[x_1 \text{ is selected}] = (1 - p_c)(1 - p_m)^l \frac{F(x_1)}{\sum\limits_{y \in Y} F(y)}. \tag{3.21}$$

If x_2 is an offspring of x_1 that undergoes crossover with probability p_c, undergoes mutation of k bits with probability p_m, and is subjected to fitness-proportional selection, it becomes a member of the population for the next generation with probability

$$\Pr[x_2 \text{ is selected}] = p_c p_m^k (1 - p_m)^{l-k} \frac{F(x_2)}{\sum\limits_{y \in Y} F(y)}. \tag{3.22}$$

Thus, the ratio of the probability of selecting x_1 to the probability of selecting x_2 becomes

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{(1 - p_c)}{p_c} \frac{(1 - p_m)^k}{p_m^k} \frac{F(x_1)}{F(x_2)} = K \frac{F(x_1)}{F(x_2)}, \;\; K > 0. \tag{3.23}$$

Although the equation above is similar to (3.4), it demonstrates that in the canonical genetic algorithm using fitness-proportional selection, the ratio of selection probabilities is proportional to the fitness ratio. In (3.23), if $K = 1$ we obtain a particular case of (3.4) where $N = 1$.

With tournament selection, the probability of considering x_1 is

$$\Pr[x_1 \text{ is considered}] = \begin{cases} p_s \in (0,1), & \text{if, } \forall y \in Y,\ F(x_1) > F(y), \\ p_s(1-p_s), & \text{if } |\{y \in Y \mid F(y) > F(x_1)\}| = 1, \\ p_s(1-p_s)^2, & \text{if } |\{y \in Y \mid F(y) > F(x_1)\}| = 2, \\ \dots & \dots \\ p_s(1-p_s)^{|Y|-1}, & \text{if } |\{y \in Y \mid F(y) > F(x_1)\}| = |Y|-1. \end{cases} \tag{3.24}$$

Using $\text{rank}(x_1)$ to denote $|\{y \in Y \mid F(y) > F(x_1)\}|$, the probability that an unchanged candidate optimizer, x_1, is a member of the population for the next generation is

$$\Pr[x_1 \text{ is selected}] = (1-p_c)(1-p_m)^l p_s(1-p_s)^{\text{rank}(x_1)}. \tag{3.25}$$

If x_2 is an offspring of x_1 that undergoes crossover with probability p_c, undergoes mutation of k bits with probability p_m, and is subjected to tournament selection, it becomes a member of the population for the next generation with probability

$$\Pr[x_2 \text{ is selected}] = p_c p_m^k (1-p_m)^{l-k} p_s(1-p_s)^{\text{rank}(x_2)}. \tag{3.26}$$

Thus, the ratio of the probability of selecting x_1 to the probability of selecting x_2 becomes

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{(1-p_c)}{p_c}\frac{(1-p_m)^k}{p_m^k}\frac{(1-p_s)^{\text{rank}(x_1)}}{(1-p_s)^{\text{rank}(x_2)}} = \\ K(1-p_s)^{\text{rank}(x_1)-\text{rank}(x_2)},\ K > 0. \tag{3.27}$$

This result is different from (3.4).

With truncation selection, the probability of considering x_1 is

$$\Pr[x_1 \text{ is considered}] = \operatorname{ind}\left(\operatorname{rank}(x_1) < \frac{|Y|}{2}\right). \tag{3.28}$$

Hence, the probability that an unchanged candidate optimizer, x_1, is a member of the population for the next generation is

$$\Pr[x_1 \text{ is selected}] = (1-p_c)(1-p_m)^l \operatorname{ind}\left(\operatorname{rank}(x_1) < \frac{|Y|}{2}\right). \tag{3.29}$$

If x_2 is an offspring of x_1 that undergoes crossover with probability p_c, undergoes mutation of k bits with probability p_m, and is subjected to truncation selection, it becomes a member of the population for the next generation with probability

$$\Pr[x_2 \text{ is selected}] = p_c p_m^k (1-p_m)^{l-k} \operatorname{ind}\left(\operatorname{rank}(x_2) < \frac{|Y|}{2}\right). \tag{3.30}$$

Thus, the ratio of the probability of selecting x_1 to the probability of selecting x_2 becomes

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{(1-p_c)}{p_c}\frac{(1-p_m)^k}{p_m^k}\frac{\operatorname{ind}\left(\operatorname{rank}(x_1) < \frac{|Y|}{2}\right)}{\operatorname{ind}\left(\operatorname{rank}(x_2) < \frac{|Y|}{2}\right)}, \tag{3.31}$$

$$= K\frac{\operatorname{ind}\left(\operatorname{rank}(x_1) < \frac{|Y|}{2}\right)}{\operatorname{ind}\left(\operatorname{rank}(x_2) < \frac{|Y|}{2}\right)},\ K > 0. \tag{3.32}$$

The ratio in (3.32) is taken to be ∞ if the denominator is zero. This ratio is different from (3.4).

3.2.4 Evolutionary Strategies

Like genetic algorithms, the general evolutionary strategy [24] operates on a population of cells of the search space of the objective function. Typically, a parent population of size μ creates an offspring population of size λ using crossover and mutation processes, with ρ parent cells required to produce one offspring. Crossover does not occur if $\rho = 1$. Selection of the population for the next generation of the algorithm occurs by picking the μ best cells from the λ offspring (known as the $(\mu/\rho, \lambda)$-ES strategy), or by picking the μ best cells from the total population of $\mu + \lambda$ cells (known as the $(\mu/\rho + \lambda)$-ES strategy).

The $(1 + 1)$-ES strategy [117] is most similar to evolutionary generation systems theory. Here, one candidate optimizer, x_1, produces one mutated offspring candidate optimizer, x_2, and the ratio of the probability of selecting x_1 to the probability of selecting x_2 is simply

$$\frac{\Pr[x_1 \text{ is selected}]}{\Pr[x_2 \text{ is selected}]} = \frac{\operatorname{ind}\left(F\left(x_1\right) > F\left(x_2\right)\right)}{\operatorname{ind}\left(F\left(x_2\right) \geq F\left(x_1\right)\right)}. \tag{3.33}$$

The ratio in (3.33) is taken to be ∞ if the denominator is zero. This ratio equals (3.4) when the parameter N in (3.4) approaches ∞.

CHAPTER 4

MARKOV CHAIN ANALYSIS OF SELECTIVE EVOLUTIONARY GENERATION SYSTEMS

4.1 Markov Chains That Behave Rationally

This section develops a Theory of Rational Behavior [29] for time-homogeneous, irreducible, ergodic Markov chains. We then discuss the entropy, resilience and opportunism of Markov chains that satisfy the axioms of this theory.

4.1.1 Markov Chain Rational Behavior

Let $(X, \mathbf{P})$ be a time-homogeneous, irreducible, ergodic Markov chain, where $X = \{x_1, x_2, \ldots, x_n\}$ is the set of states of a Markov process, $\mathbf{P} \in \mathbb{R}^{n \times n}$ is the matrix of transition probabilities for these states, and $n < \infty$ is the number of states. Assume that the initial probability distribution over the states is known, i.e., we are given an n-vector $\mathbf{p}(0)$ having elements $p_i(0) = \Pr[\mathcal{X}(0) = x_i]$ for all $x_i \in X$, where $\mathcal{X}(0)$ denotes the state realization at time 0, and we have $\sum_{i=1}^{n} p_i(0) = 1$. Since we have assumed that the states in X are ergodic and irreducible, they admit a unique stationary probability distribution [113, 114]. Let $\boldsymbol{\pi} = \begin{bmatrix} \pi_1 & \pi_2 & \ldots & \pi_n \end{bmatrix}$ be the row vector of these stationary probabilities, satisfying the constraints $\pi_i > 0 \; \forall i$, and $\sum_{i=1}^{n} \pi_i = 1$. Let $F : X \to \mathbb{R}^+$ be a positive fitness function. Let $N \in \mathbb{N}$ be a natural number. We define rational behavior for this Markov chain as follows.

Definition 4.1. The time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ is said to *behave rationally* with respect to fitness F with level N if

$$\frac{\pi_i}{\pi_j} = \left(\frac{F(x_i)}{F(x_j)}\right)^N, \; 1 \leq i, j \leq n. \tag{4.1}$$

This definition of *global rationality* is consistent with [29] because time averages and ensemble averages are equal in an ergodic process. The requirement that $\pi_i > 0 \; \forall i$ with $\sum\limits_{i=1}^{n} \pi_i = 1$ corresponds to the ergodic postulate of [29], and the requirement that $N > 0$ corresponds to the selective (i.e., retardation) postulate. Note that we have recast the requisite scalar function of [29] as a reward, instead of a penalty.

Each stationary probability can also be explicitly characterized to ensure Markov chain rational behavior, as is indicated by the following theorem.

Theorem 4.1. *The time-homogeneous, irreducible, ergodic Markov chain* $(X, \mathbf{P})$ *behaves rationally with respect to fitness* F *with level* N *if and only if*

$$\pi_i = \frac{F(x_i)^N}{\sum\limits_{k=1}^{n} F(x_k)^N}, \; 1 \leq i \leq n. \tag{4.2}$$

Proof. See Appendix A. □

Here, we have a more general, probabilistic version of the optimization of an objective function. A Markov chain that behaves rationally will select the state of maximum fitness with the highest stationary probability, and, in the limit as N approaches ∞, this probability is 1. The problem and solution then revert to one of standard optimization.

Remarkably, rational behavior in Markov chains is the result of a subsidiary optimization.

Theorem 4.2. *The stationary distribution* $\boldsymbol{\pi}$ *of the time-homogeneous, irreducible, ergodic Markov chain* $(X, \mathbf{P})$ *that behaves rationally with respect to fitness* F *with* *level N solves the optimization problem*

$$\min_{\pi_1,\ldots,\pi_n} \Phi(\boldsymbol{\pi}) = -\sum_{i=1}^{n} F(x_i)^N \ln(\pi_i), \tag{4.3}$$

subject to the constraints

$$\sum_{i=1}^{n} \pi_i = 1, \tag{4.4}$$

$$\pi_i > 0, \ \forall i. \tag{4.5}$$

Proof. See Appendix A. □

Note that in (4.1), rational behavior is invariant under positive scaling of fitness. Hence, there is no loss of generality in assuming that the fitness function is normalized. Accordingly, let $\boldsymbol{\varphi} = \begin{bmatrix} \varphi_1 & \varphi_2 & \ldots & \varphi_n \end{bmatrix}$ be the distribution of the N^{th} power of fitness, where

$$\varphi_i = \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N}, \ 1 \leq i \leq n. \tag{4.6}$$

Definition 4.2. A vector $v \in \mathbb{R}^n$ is a *positive mass function of order* n if it satisfies $v_i > 0 \ \forall i$, and $\sum\limits_{k=1}^{n} v_k = 1$. Let $\mathbb{D}_n$ be the set of positive mass functions of order n.

The vector $\boldsymbol{\varphi} \in \mathbb{R}^n$ is a positive mass function. Let

$$U(\boldsymbol{\pi}) = \frac{\Phi(\boldsymbol{\pi})}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N}. \tag{4.7}$$

Then, the optimization problem (4.3) can be normalized as

$$\min_{\pi_1,\ldots,\pi_n} U(\boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln(\pi_i), \tag{4.8}$$

subject to the constraints (4.4) and (4.5). Furthermore, Theorem 4.2 states that at the optimum, the stationary distribution agrees with the fitness distribution, i.e., $\boldsymbol{\pi} = \boldsymbol{\varphi}$.

4.1.2 Entropy of Markov Chains That Behave Rationally

Definition 4.3. *Entropy* [118] is the function

$$H : \mathbb{D}_n \to \mathbb{R} : \boldsymbol{\varphi} \mapsto H(\boldsymbol{\varphi}) = -\sum_{i=1}^{n} \varphi_i \ln(\varphi_i). \tag{4.9}$$

Using the notion of entropy, we can interpret (4.8) as follows. First, we recognize the term $-\ln(\pi_i)$ as the information content of state x_i [118]. Hence, the right hand side of (4.8) represents the "fitness-expectation of information." Moreover, we have the following:

Corollary 4.1. *The time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ behaves rationally with respect to fitness F with level N if and only if its stationary probability distribution minimizes the fitness-expectation of information. At the optimum, this fitness-expectation of information is the entropy of the fitness distribution, i.e.,*

$$U^* = H(\boldsymbol{\varphi}) = -\sum_{i=1}^{n} \varphi_i \ln(\varphi_i). \tag{4.10}$$

A basic property of entropy that is alluded to in [119] and which will be utilized in the proof of Theorem 4.12 follows.

Theorem 4.3. *Let $\boldsymbol{\varphi} \in \mathbb{D}_n$ be arbitrary. Then,*

$$\min_{\boldsymbol{\pi} \in \mathbb{D}_n} -\sum_{i=1}^{n} \varphi_i \ln(\pi_i), \tag{4.11}$$

has a minimum value of $H(\boldsymbol{\varphi})$ that is achieved at $\boldsymbol{\pi} = \boldsymbol{\varphi}$.

Equivalently, $\forall \boldsymbol{\varphi} \in \mathbb{D}_n$, $\forall \boldsymbol{\pi} \in \mathbb{D}_n$,

$$-\sum_{i=1}^{n} \varphi_i \ln(\pi_i) \geq -\sum_{i=1}^{n} \varphi_i \ln(\varphi_i), \tag{4.12}$$

with the equality holding if and only if $\boldsymbol{\pi} = \boldsymbol{\varphi}$.

Equivalently, $\forall \boldsymbol{\varphi} \in \mathbb{D}_n$, $\boldsymbol{\pi} \in \mathbb{D}_n$,

$$-\sum_{i=1}^{n} \varphi_i \ln \left(\frac{\pi_i}{\varphi_i} \right) \geq 0, \tag{4.13}$$

with the equality holding if and only if $\boldsymbol{\pi} = \boldsymbol{\varphi}$.

Proof. See Appendix A. □

For Markov chains that behave rationally, and therefore possess fitness fractions that are distributed over the set of states as in (4.6), the entropy quantifies how egalitarian or elitist the states are. That is, the entropy is highest when all states have equal fitness; conversely, the entropy is lowest when there is only one state with a fitness fraction of unity and all other fitness fractions are zero. Equation (4.9) arises in other well-known fields, and similar interpretations for the distributed quantities and the entropy exist — see Table 4.1.

In Information Theory [118], the distributed quantity is the probability that a particular message is chosen given the sequence of symbols received, and this probability is distributed over the set of all messages. Entropy in this theory quantifies certainty and uncertainty. Equation (4.8) is also derived and discussed in [119] within the context of Information Theory. In Classical Thermodynamics [120], the mole fractions of a physical/chemical substance are distributed over the set of all physical/chemical substances in the system. Entropy in this theory quantifies separation and mixing. In Statistical Mechanics [121], the distributed quantity is the probability that a system is in a particular microstate given its macrostate, and this probability is distributed over the set of all microstates in the system. Entropy in this theory quantifies order and disorder.

Table 4.1: Entropy in the Sciences

	Theory of Rational Behavior	Information Theory [118]	Classical Thermodynamics [120]	Statistical Mechanics [121]
Distributed Quantity	Fitness fraction of Markov states	Probability that a particular message is chosen given the sequence of symbols received	Mole fraction of a physical/chemical substance	Probability that a system is in a particular microstate given its macrostate
Space That the Quantity is Distributed Over	Set of states of a Markov process	Set of messages	Set of physical/ chemical substances in the system	Set of microstates
Entropy Quantified Motion	Elitism vs. egalitarianism	Certainty vs. uncertainty[a]	Separation vs. mixing	Order vs. disorder

---

[a] Equation (4.8) is also derived and discussed in [119] within the context of Information Theory.

Section 4.2.2 considers a discrete-time dynamic system described by

$$\mathbf{p}(t+1) = \mathbf{p}(t)\mathbf{P}, \tag{4.14}$$

where the underlying dynamics are given by a time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ that behaves rationally with respect to fitness F with level N, and $\mathbf{p}(t)$ is an n-dimensional row vector at time t. It is shown that the discrete-time dynamic system has an invariant manifold, which is the set of vectors $\mathbf{p}$ with components $p_i(t) > 0,\ 1 \leq i \leq n$, and $\sum_{i=1}^{n} p_i(t) = 1$. It is also shown that the manifold has an equilibrium for these dynamics, $\boldsymbol{\pi}$, with components π_i satisfying (4.2), and that the function

$$V(\mathbf{p}(t)) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{p_i(t)}{\varphi_i}\right), \tag{4.15}$$

where φ_i satisfies (4.6), is a Lyapunov function that establishes global asymptotic stability of the dynamic system (4.14) with respect to the manifold. Equation (4.15) can be defined for any Markov chain that behaves rationally, since it is the non-stationary and non-biased version of (4.7). However, (4.15) has deeper meanings: a relationship to stability analysis for the technique in this dissertation, and as we shall see, a relationship to search theory too.

Reference [122] provides a relationship between search theory, (4.15) and a similar expression

$$W(\mathbf{p}(t)) = -\sum_{i=1}^{n} p_i(t) \ln\left(\frac{p_i(t)}{\varphi_i}\right). \tag{4.16}$$

Initial values $V(\mathbf{p}(0))$ and $W(\mathbf{p}(0))$ are measures of prior information for a search. It is stated that $V(\mathbf{p}(t)) - V(\mathbf{p}(0))$ is 'the measure of the amount of prior information utilized' by the search up to time t, and $W(\mathbf{p}(t)) - W(\mathbf{p}(0))$ is 'the measure of the savings in search effort thereby achieved. The optimal (search) policy is then the one that trades off initial information for reduced search effort, as quickly as possible' [122].

Entropy maximization is important for search according to [123] as follows.

> ‘In making inferences on the basis of partial information, the maximum entropy probability distribution subject to whatever is known is the only unbiased assignment we can make; to use any other would amount to arbitrary assumption of information which by hypothesis we do not have. Mathematically, the maximum-entropy distribution has the important property that no possibility is ignored; it assigns a positive weight to every situation that is not absolutely excluded by the given information. This is quite similar in effect to an ergodic property.’

The relationship between entropy maximization and optimal search is clarified in [122]. The optimal search policy for cells with exponential “sizes” ‘appears very much like an irreversible process in thermodynamics, in which an initially non-equilibrium state relaxes in the the equilibrium state of maximum entropy. But now it is only our state of knowledge that relaxes to the “equilibrium” condition of maximum uncertainty’ [122].

Applying these results from [122] and [123], an exponential normalized fitness function relates rational behavior, entropy and optimal search through the following theorem.

Theorem 4.4. *Let* $y : X \to \mathbb{R}$ *be an unknown function for which an expected value,* $\mathrm{E}\left[y(x)\right]$*, is a known number* Y*. The normalized fitness*

$$\varphi_i = \alpha e^{-\beta y(x_i)},\ 1 \leq i \leq n, \tag{4.17}$$

and the stationary distribution $\boldsymbol{\pi}$ *of the time-homogeneous, irreducible, ergodic Markov chain* $(X, \mathbf{P})$ *that behaves rationally with respect to fitness* F *with level* N *solves the* *optimization problem*

$$\max_{\boldsymbol{\varphi} \in \mathbb{D}_n} \min_{\boldsymbol{\pi} \in \mathbb{D}_n} U(\boldsymbol{\varphi}, \boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln(\pi_i), \tag{4.18}$$

subject to the constraint

$$\mathrm{E}\left[y(x)\right] = Y. \tag{4.19}$$

Proof. See Appendix A. □

Hence, a scheme with underlying Markov chain dynamics that behave rationally also maximizes the entropy of the fitness distribution when the fitness function is exponential. The implication is that a fitness function like

$$F(x_i) = e^{-\left((z(x_i)-Z)^2\right)} \tag{4.20}$$

together with a scheme that makes use of rational behavior guarantees good behaviors efficiently, in accordance with the problem definition of Section 1.4.

Exponential fitness functions arise in nature if one considers fecundity as the measure of fitness. Other examples include the beak depth of the Galapagos finches [124] and instances when *directional selection* [105] is prevalent.

4.1.3 Resilience and Opportunism of Markov Chains That Behave Rationally

We can now formally define resilience and opportunism, first described through Figure 1.1, as the sensitivity of the stationary distribution to changes in fitness.

Definition 4.4. For any time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ with a positive fitness function for all the states in X, the *extrinsic resilience* of state

x_i to changes in the fitness of state x_j, $j \neq i$, is defined as

$$\rho_{ij} = \frac{\partial \pi_i}{\partial F(x_j)}, \tag{4.21}$$

and the *intrinsic resilience* of state x_i to changes in its own fitness is taken to be

$$\rho_{ii} = \frac{\partial \pi_i}{\partial F(x_i)}. \tag{4.22}$$

Since the stationary distribution $\boldsymbol{\pi}$ has the closed form expression (4.2) for the time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ that behaves rationally with respect to fitness F with level N, the extrinsic and intrinsic resiliencies are

$$\rho_{ij} = \frac{\partial \pi_i}{\partial F(x_j)} = \frac{-N\pi_i\pi_j}{F(x_j)}, \ \forall j \neq i, \tag{4.23}$$

$$\rho_{ii} = \frac{\partial \pi_i}{\partial F(x_i)} = \frac{N\pi_i\,(1-\pi_i)}{F(x_i)}. \tag{4.24}$$

We say that the Markov chain $(X, \mathbf{P})$ is *resilient and opportunistic* if $\rho_{ij} \neq 0$ for all i and j.

The level of selectivity has the following asymptotic effect on resilience and opportunism.

Theorem 4.5. *For the time-homogeneous, irreducible, ergodic Markov chain* $(X, \mathbf{P})$ *that behaves rationally with respect to fitness* F *with level* N*,*

$$\rho_{ij}\Big|_{\substack{N=0\\ j\neq i}} = \rho_{ii}\Big|_{N=0} = 0, \tag{4.25}$$

and

$$\lim_{\substack{N\to\infty\\ j\neq i}} \rho_{ij} = \lim_{N\to\infty} \rho_{ii} = 0. \tag{4.26}$$

Proof. See Appendix A. □

As a result of Theorem 4.5, we have quantification that standard optimization ($N \to \infty$) is non-resilient. Moreover, recall that if we assume a k-generation fecundity interpretation of fitness as in Section 3.1, then $N \to \infty$ also represents an infinite number of selections made over k generations. There is much biological evidence to confirm that prolonged selective breeding yields non-resilient strains [125–129].

Resilience and opportunism is a direct outcome of Markov chain rational behavior, as stated below.

Theorem 4.6. *The time-homogeneous, irreducible, ergodic Markov chain* $(X, \mathbf{P})$ *is resilient and opportunistic if the chain behaves rationally.*

Proof. See Appendix A. □

Resilience and opportunism do not always imply Markov chain rational behavior (see Section 4.3). But we can state the following instead.

Theorem 4.7. *Ergodicity is a necessary condition for the time-homogeneous, irreducible Markov chain* $(X, \mathbf{P})$ *to be resilient and opportunistic.*

Proof. See Appendix A. □

Furthermore, there is a fundamental trade-off between extrinsic and intrinsic resilience that is imposed by the constraint $\sum_{i=1}^{n} \pi_i = 1$. Taking the partial derivative of this constraint with respect to the fitness of state x_i, we obtain

$$\frac{\partial \pi_i}{\partial F(x_i)} + \sum_{\substack{j=1 \\ j \neq i}}^{n} \frac{\partial \pi_j}{\partial F(x_i)} = 0. \tag{4.27}$$

Note that, from (4.23) and (4.24), the extrinsic resiliencies are always negative, whereas the intrinsic resiliencies are positive. Hence, (4.27) implies that any change in fitness that improves a state's intrinsic resilience is at the expense of the extrinsic resilience of all other states. Similarly, any change in fitness that improves a state's extrinsic resilience is at the expense of the intrinsic resilience of another state, and the extrinsic resilience of all other states.

4.1.4 The Importance of Resilience and Opportunism

The results of Sections 4.1.2 and 4.1.3 facilitate a qualitative discussion of the domains where resilience and opportunism are important. Such a discussion is the focus of this subsection, and the discussion is partly inspired by the premise of [119] and the interpretation of (4.7) therein.

When searching for a desirable behavior, a fit species, or an objective function optimizer, the search is dependent on two features: the desirability, fitness or optimality of the elements of the search space, and the definition of desirability, fitness, or objective function to be optimized. Accordingly, there are two types of uncertainties associated with the search. The first is an *uncertainty in belief* about element suitability, which is a vagueness that is due to a lack of information about the search space. This uncertainty is embodied in a prior probability distribution on the space of actions, organisms, or candidate optimizers, and is reduced when the distribution changes to a posterior probability distribution after an informative search. Information Theory [118] deals with this kind of uncertainty.

The second type of uncertainty is an *uncertainty in environment*, which stems from uncertain definitions of desirability, fitness, or objective functions. This uncertainty is caused by the possibility that the possessed information about the search space is incorrect, since changes in definition could lead to changes in element suitability. Reference [119] augmented Information Theory to include this kind of uncertainty. Note that sampling the search space and finding element suitability to be unchanged with each sample does not always resolve uncertainty in the environment. This is because element suitability may change during the interval between samples (reminiscent of the need to sample according to Shannon's sampling theorem [130]).

Both uncertainty in belief and uncertainty in environment can exist simultaneously during search. The size of the search space can also affect the choice of search technique. These three features are useful in determining domains where resilience and opportunism are important and where they are not required (see Figure 4.1).

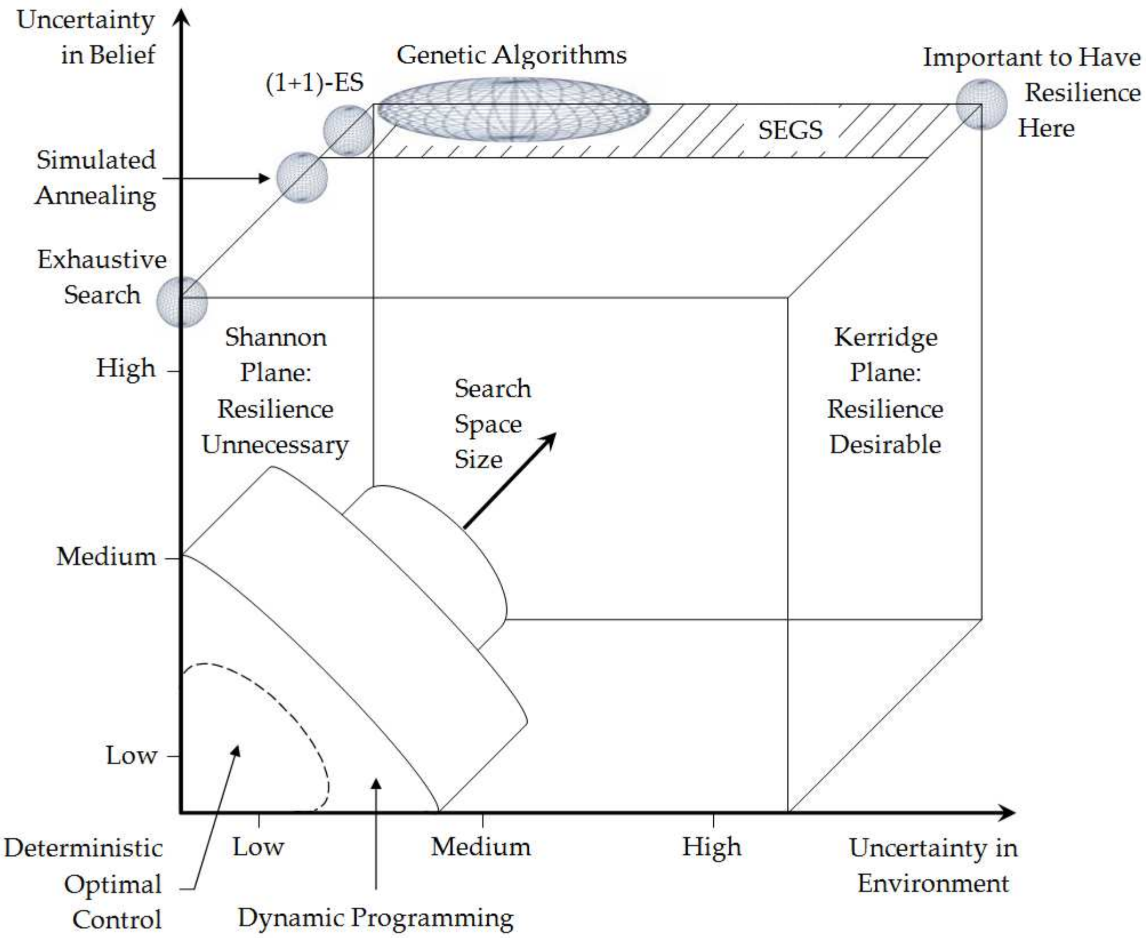


Figure 4.1: The Markov Cube depicts how common approaches to optimization compare with respect to belief and environment uncertainties, and search space size.

As the following discussion shows, resilience and opportunism are not required when the environment is certain. This corresponds to the left plane of the cube in Figure 4.1, called the Shannon plane. The right plane of the cube, called the Kerridge plane, is where resilience and opportunism are desirable, and where (4.7) arises. Resilience and opportunism are most important in the worst case scenario, when belief uncertainty, environment uncertainty, and search space size are all high. The results of Section 4.1.3 indicate that Markov chains that behave rationally start at the Shannon plane when level $N = 0$, move towards the Kerridge plane for intermediate values of level N, and then traverse back to the Shannon plane as level N approaches ∞.

Consider a domain where the search space has few elements, the environment is certain (i.e., there is no possibility of changes in what is considered desirable, fit or optimal), and there is no prior information about the desirability, fitness or optimality of the elements. An exhaustive search of the small space suffices to eliminate high uncertainties in belief. Resilience is not required for this domain.

An example of a domain where both environment and belief uncertainties are low is the field of deterministic optimal control theory [131–133], which is used to find a control strategy for a dynamic system that maximizes a performance measure. Typically, this performance measure is specified and external disturbances are ignored, resulting in low environment uncertainty. Although the space of admissible control inputs may be large, prior information about the system (and therefore the feasible control inputs) exists because the dynamics are known and the outputs are presumed observable. Hence, there is low belief uncertainty. Resilience is not important in this setting because of the low uncertainties.

The more general stochastic optimal control problem [134, 135] includes external disturbances and partial observability. Partial observability of the system increases the uncertainty in belief about the system states and the appropriate control inputs. The performance measure to be maximized is specified explicitly (as an expectation), and statistics about the disturbances are typically assumed. Thus, environment uncertainty is still relatively low for this domain. Dynamic programming is the technique that is often used to solve stochastic optimal control problems. Resilience is not required because dynamic programming yields the optimal control inputs for all possible states of the system. Consequently, dynamic programming suffers from the curse of dimensionality, and is limited to smaller search space sizes.

The multi-armed bandit problem [136] (with application to reinforcement learning [16]) is a similar instance of a domain with moderate belief uncertainty and low environment uncertainty. Probabilistic payoffs result in medium uncertainty in belief, while the fixed payoff distributions result in low uncertainty in environment. The optimal Gittins index policy can be obtained through dynamic programming, and resilience is therefore not required.

The remaining techniques of Section 3.2 can also be placed on the cube in Figure 4.1. These methods are appropriate for large search spaces and high belief uncertainty. Simulated annealing is located on the Shannon plane because the designed behavior is not resilient once the parameter T has been reduced. Genetic algorithms, on the other hand, can be resilient; this is suggested by comparing the canonical genetic algorithm with fitness proportional selection to a Markov chain that behaves rationally with level $N = 1$. The (1+1) evolutionary strategy is located on the Shannon plane because it can be compared to a Markov chain that behaves rationally with level $N \to \infty$.

Section 4.2 describes how a stochastic SEGS process is a Markov chain that behaves rationally with level N between 0 and ∞. The proposed scheme in this dissertation occupies the hatched area of the cube in Figure 4.1 because a SEGS is also suited for high belief uncertainty and moderate to large search spaces (contingent on the number of parallel implementations). It follows that a SEGS generalizes genetic algorithms and the (1+1) evolutionary strategy as expected.

Since Markov decision processes and Markov chains can be used to analyze the techniques depicted in Figure 4.1, the cube in the figure is called a Markov Cube.

4.2 Selective Evolutionary Generation Systems as Markov Chains That Behave Rationally

This section applies the Theory of Rational Behavior for time-homogeneous, irreducible, ergodic Markov chains (as developed in Section 4.1) to a SEGS as formulated in Section 3.1. We begin with some preliminaries.

4.2.1 Analysis of Selective Evolutionary Generation Systems

Definition 4.5. Let $\Gamma = (X, R, P, G, F)$ be a selective evolutionary generation system. Let $x_i, x_j \in X$ be any two cells, and $r_k \in R$ be a resource. The *descendancy tensor*, δ, has elements

$$\delta_{ijk} = \begin{cases} 1 & \text{if } x_j = G(x_i, r_k),\ 1 \le i \le n,\ 1 \le j \le n,\ 1 \le k \le m, \\ 0 & \text{otherwise.} \end{cases} \tag{4.28}$$

Hence, the descendancy tensor indicates whether it is possible to produce cell x_j in one step from cell x_i, using resource r_k. We can use this tensor to create a matrix that represents the conditional probability of generating x_j given that the progenitor is x_i, by utilizing the probability of selecting each available resource and summing over all m resources as follows.

Definition 4.6. For the SEGS $\Gamma = (X, R, P, G, F)$, the *matrix of generation probabilities*, $\boldsymbol{\gamma}$, also called the unselective matrix of transition probabilities, has elements

$$\begin{aligned} \gamma_{ij} &= \Pr[\text{offspring is } x_j \mid \text{progenitor is } x_i], && (4.29) \\ &= \sum_{k=1}^{m} \delta_{ijk} p_k,\ 1 \le i \le n,\ 1 \le j \le n. && (4.30) \end{aligned}$$

This matrix is a stochastic matrix, as indicated by the following lemma.

Lemma 4.1. *For the SEGS* $\Gamma = (X, R, P, G, F)$ *with matrix of generation probabilities* $\boldsymbol{\gamma}$*,*

$$\sum_{j=1}^{n} \gamma_{ij} = 1, \ 1 \le i \le n. \tag{4.31}$$

Proof. See Appendix A. □

Recall that a SEGS follows the stochastic Markov process described by (3.3). Therefore, we can find a matrix of transition probabilities to describe the cell-to-cell transitions that occur as a result of the selection dynamics. For the SEGS $\Gamma = (X, R, P, G, F)$, the *matrix of transition probabilities*, $\mathbf{P}$, has elements

$$\begin{aligned} P_{ij} =& \Pr[\mathcal{X}(t+1) = x_j \mid \mathcal{X}(t) = x_i], && (4.32)\\ =& \Pr[\mathit{Select}(x_i, x_j, N) = x_j \mid \mathcal{X}(t) = x_i] \times \\ & \Pr[\text{offspring is } x_j \mid \text{progenitor is } x_i] && (4.33)\\ =& \begin{cases} \frac{1}{1+\left(\frac{F(x_i)}{F(x_j)}\right)^N} \gamma_{ij}, & \forall j \ne i, \\ \gamma_{ii} + \sum\limits_{\substack{j=1\\ j\ne i}}^{n} \frac{1}{1+\left(\frac{F(x_j)}{F(x_i)}\right)^N} \gamma_{ij}, & \text{if } j = i. \end{cases} && (4.34) \end{aligned}$$

Note that the matrix of transition probabilities in (4.34) is also a stochastic matrix.

Theorem 4.8. *For the SEGS* $\Gamma = (X, R, P, G, F)$ *with matrix of transition probabilities* $\mathbf{P}$*,*

$$\sum_{j=1}^{n} P_{ij} = 1, \ 1 \le i \le n. \tag{4.35}$$

Proof. See Appendix A. □

In addition to irreducibility, if we assume that the selection dynamics of the SEGS is ergodic, then a unique stationary probability distribution over the set of cells exists, and must satisfy the following.

Theorem 4.9. *For the ergodic SEGS* $\Gamma = (X, R, P, G, F)$*, let* $\boldsymbol{\pi} = \begin{bmatrix} \pi_1 & \pi_2 & \ldots & \pi_n \end{bmatrix}$ *be the row vector of stationary probabilities, satisfying* $\sum_{i=1}^{n} \pi_i = 1$*. Assume that there is a unique index,* I*, such that* $F(x_i)$ *is maximized for* $i = I$*. Then,*

$$\lim_{N\to\infty} \pi_i = 0,\ 1 \le i \le n,\ i \ne I, \tag{4.36}$$

$$\lim_{N\to\infty} \pi_I = 1. \tag{4.37}$$

Proof. See Appendix A. □

It is easy to extend this theorem and its proof to the case where I is not unique and show that the cells with equal maximal fitness are equiprobable. For both versions of the theorem, there exist stationary probabilities equal to zero in the limit as N approaches ∞ because the stochastic selection process becomes elitist instead of ergodic.

We can also examine the SEGS response to changes in selectivity and cell fitness. First, the probability of increasing fitness with every time step, conditioned upon knowledge of the current cell, is

$$\begin{aligned} &\Pr\left[F\left(\mathcal{X}\left(t+1\right)\right) > F\left(\mathcal{X}\left(t\right)\right) \mid \mathcal{X}\left(t\right) = x_i\right] \\ =&\sum_{\substack{j=1\\ j\ne i}}^{n} \operatorname{ind}\left(F\left(x_j\right) > F\left(x_i\right)\right) P_{ij}. \end{aligned} \tag{4.38}$$

This conditional probability increases as N increases. However, the unconditional probability of increasing fitness with every time step,

$$\begin{aligned} &\Pr\left[F\left(\mathcal{X}\left(t+1\right)\right) > F\left(\mathcal{X}\left(t\right)\right)\right] \\ =&\sum_{i=1}^{n}\sum_{\substack{j=1\\ j\ne i}}^{n} \operatorname{ind}\left(F\left(x_j\right) > F\left(x_i\right)\right) P_{ij}\pi_i, \end{aligned} \tag{4.39}$$

approaches zero in the limit as N approaches ∞. That is, the unconditional probability decreases as N increases. This (perhaps counter-intuitive) result is due to the elitist nature of the resultant selection process — the cell with maximal fitness has a stationary probability of 1, and consequently, the probability of improving fitness is correspondingly 0.

Next, the effect of changes in cell fitness on elements of the matrix of transition probabilities, $\mathbf{P}$, is given by the following four equations:

$$\begin{aligned}
\forall j \neq i, \frac{\partial P_{ij}}{\partial F(x_j)} &= \frac{N}{F(x_j)} \left(\frac{1}{1 + \left(\frac{F(x_j)}{F(x_i)}\right)^N} \right) P_{ij}, && (4.40)\\
\frac{\partial P_{ii}}{\partial F(x_j)} &= \frac{-N}{F(x_j)} \sum_{\substack{j=1\\ j\neq i}}^{n} \frac{1}{1 + \left(\frac{F(x_j)}{F(x_i)}\right)^N} P_{ij}, && (4.41)\\
\forall j \neq i, \frac{\partial P_{ij}}{\partial F(x_i)} &= \frac{-N}{F(x_i)} \left(\frac{1}{1 + \left(\frac{F(x_j)}{F(x_i)}\right)^N} \right) P_{ij}, && (4.42)\\
\frac{\partial P_{ii}}{\partial F(x_i)} &= \frac{N}{F(x_i)} \sum_{\substack{j=1\\ j\neq i}}^{n} \frac{1}{1 + \left(\frac{F(x_j)}{F(x_i)}\right)^N} P_{ij}. && (4.43)
\end{aligned}$$

In the first equation above, we see that an increase in the fitness of cell x_j increases the probability of transitioning to that cell from current cell x_i by an amount that is proportional to the level of selectivity and inversely proportional to the fitness value. The second equation indicates a corresponding decrease in the probability of transitioning back to the current cell under the same altered fitness landscape. Unlike gradient ascent optimization where the transition to another cell would be directly proportional to the fitness value, what we have here is reminiscent of the retardation property in [29]; the stochastic process "slows down" transitions in more favorable fitness conditions to take advantage of the external environment. Similar effects on the transition probabilities are suggested by the latter two equations for changes in current cell fitness.

4.2.2 Dynamic Properties of Selective Evolutionary Generation Systems

We can now state some dynamic properties of selective evolutionary generation systems, under certain technical conditions.

Theorem 4.10. *For the ergodic SEGS* $\Gamma = (X, R, P, G, F)$*, assume that the matrix of generation probabilities,* $\boldsymbol{\gamma}$*, is symmetric. Then the Markov chain representing the stochastic dynamics of the ergodic SEGS behaves rationally with fitness* F *and level* N*. That is, the row vector* $\boldsymbol{\pi} = \begin{bmatrix} \pi_1 & \pi_2 & \dots & \pi_n \end{bmatrix}$*, where* π_i *satisfies* (4.2)*, is a left eigenvector of* $\mathbf{P}$*, the matrix of transition probabilities for* Γ*, with corresponding eigenvalue 1 (i.e.,* $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$*). Hence,* $\boldsymbol{\pi}$ *is the vector of stationary probabilities for the SEGS.*

Proof. See Appendix A. □

As a result of Theorem 4.6, the stochastic dynamics of the ergodic SEGS with symmetric matrix of generation probabilities, $\boldsymbol{\gamma}$, are resilient and opportunistic. Hence, a SEGS is a computationally inexpensive on-line technique to achieve these characteristics because only local decisions between two candidate optimizers are made at any time. The need to evaluate the fitness of all elements in the domain of the objective function, or even in a sub-population of candidate optimizers (as in genetic algorithms or evolutionary strategies), is avoided.

The symmetry condition on the matrix of generation probabilities, $\boldsymbol{\gamma}$, implies that there exists equiprobable forward and reverse transitions between any pair of cells prior to the selection process. More specifically, symmetry of $\boldsymbol{\gamma}$ is a requirement that mutations be reversible. This reversibility requirement is satisfied in biology, and such mutations are called *true back mutations* [137, 138].

Theorem 4.11. *For the ergodic SEGS* $\Gamma = (X, R, P, G, F)$*, assume that the matrix of generation probabilities,* $\boldsymbol{\gamma}$*, is symmetric. Then the Markov chain representing the* *stochastic dynamics of the ergodic SEGS is time-reversible, i.e.,*

$$\pi_i P_{ij} = \pi_j P_{ji}, \ \forall i, j. \tag{4.44}$$

Proof. See Appendix A. □

As a consequence, the Markov chain representing the stochastic dynamics of the SEGS and its time reversed form are statistically the same.

Theorem 4.12. *For the ergodic SEGS* $\Gamma = (X, R, P, G, F)$*, assume that the matrix of generation probabilities,* $\boldsymbol{\gamma}$*, is symmetric. Consider the discrete-time dynamic system described by*

$$\mathbf{p}(t+1) = \mathbf{p}(t)\mathbf{P}, \tag{4.45}$$

where $\mathbf{P}$ *is the matrix of transition probabilities for* Γ*, and* $\mathbf{p}(t)$ *is an* n*-dimensional row vector at time* t*.*

(1) This discrete-time dynamic system has an invariant manifold. The manifold is the set of vectors $\mathbf{p}$ *with components* $p_i(t) > 0, \ 1 \leq i \leq n$*, and* $\sum_{i=1}^{n} p_i(t) = 1$*.*

(2) The manifold has an equilibrium for these dynamics, $\boldsymbol{\pi}$*, with components* π_i *satisfying* (4.2)*.*

(3) The function

$$V(\mathbf{p}(t)) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{p_i(t)}{\varphi_i}\right), \tag{4.46}$$

where φ_i *satisfies* (4.6)*, is a Lyapunov function that establishes global asymptotic stability of the dynamic system* (4.45) *with respect to the manifold.*

Proof. See Appendix A. □

Another important quantity of an ergodic SEGS is the expected amount of time to reach the fittest cell, given a starting cell. We will make use of the following related definitions, which are common to the theory of Markov chains [114].

Definition 4.7. The *return time* T_j to cell x_j is

$$T_j = \inf\{t \geq 1 \mid \mathcal{X}(t) = x_j\}, \tag{4.47}$$

where $T_j = \infty$ if $\mathcal{X}(t) \neq x_j$ for all $t \geq 1$.

The *hitting time* of x_j is taken to be

$$S_j = \begin{cases} T_j, & \text{if } \mathcal{X}(0) \neq x_j, \\ 0, & \text{if } \mathcal{X}(0) = x_j. \end{cases} \tag{4.48}$$

The *mean hitting time* to x_j given an initial cell x_i is defined as

$$\sigma_{ij} = \mathrm{E}\left[S_j \mid \mathcal{X}(0) = x_i\right] = \begin{cases} \sum\limits_{k=1}^{n} P_{ik}\left(\sigma_{kj} + 1\right), & \forall i \neq j, \\ 0, & i = j, \end{cases} \tag{4.49}$$

and we take $\boldsymbol{\sigma}_j = \begin{bmatrix} \sigma_{1j} & \sigma_{2j} & \dots & \sigma_{nj} \end{bmatrix}^{\mathrm{T}}$.

If we let $\mathbf{1} = \begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}^{\mathrm{T}}$ and $\mathbf{D}_j$ be a diagonal matrix with ones on the diagonal except one zero at position (j, j), then

$$\boldsymbol{\sigma}_j = \mathbf{D}_j(\mathbf{P}\boldsymbol{\sigma}_j + \mathbf{1}). \tag{4.50}$$

Alternatively,

$$\boldsymbol{\sigma}_j = (\mathbf{I} - \mathbf{D}_j\mathbf{P})^{-1}\mathbf{D}_j\mathbf{1}, \tag{4.51}$$

where $\mathbf{I}$ is the $n \times n$ identity matrix.

Theorem 4.13. *For the ergodic SEGS $\Gamma = (X, R, P, G, F)$, assume that there exists a unique index I such that $F(x_i)$ is maximized for $i = I$. Then for all $i \neq I$,*

1. *$\lim\limits_{N\to\infty} \sigma_{iI}$ exists, and*

2. *σ_{iI} is a strictly decreasing function of N.*

Proof. See Appendix A. □

Hence, a trade-off exists between resilient and opportunistic behavior of the SEGS, and the expected hitting time of the optimizer, with the trade-off controlled by the level of selectivity, N. That is, increasing N reduces the mean hitting time to the fittest cell but also decreases resilience and opportunism.

4.3 Illustrative Example of a Selective Evolutionary Generation System

We illustrate the theory in this dissertation with an example in two parts.

4.3.1 Non-symmetric Matrix of Generation Probabilities

Consider the evolutionary generation system (X, R, P, G), where

- $X = \{x_1, x_2\}$,
- $R = \{r_1, r_2\}$,
- $P(r_1) = p$, $P(r_2) = 1 - p$, $p \neq 0$,
- $G(x_1, r_1) = x_2$, $G(x_1, r_2) = x_1$, $G(x_2, r_1) = x_2$, and $G(x_2, r_2) = x_1$ (see Figure 4.2).

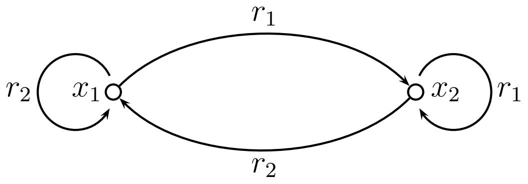


Figure 4.2: The directed graph of the example in Section 4.3.1.

The matrix of generation probabilities for this evolutionary generation system is

$$\boldsymbol{\gamma} = \begin{bmatrix} 1-\gamma_{12} & \gamma_{12} \\ \gamma_{21} & 1-\gamma_{21} \end{bmatrix} = \begin{bmatrix} 1-p & p \\ 1-p & p \end{bmatrix}. \tag{4.52}$$

Let $F(x_1) = f_1$ and $F(x_2) = f_2$. Let N be a finite level of selectivity. Utilizing (4.34), the matrix of transition probabilities for the selective evolutionary generation system is

$$\mathbf{P} = \frac{1}{f_1^N + f_2^N} \begin{bmatrix} f_1^N + (1-\gamma_{12})f_2^N & \gamma_{12}f_2^N \\ \gamma_{21}f_1^N & (1-\gamma_{21})f_1^N + f_2^N \end{bmatrix}. \tag{4.53}$$

This SEGS is both aperiodic and positive recurrent, and hence, ergodic.

The stationary distribution of the SEGS can be computed to be

$$\boldsymbol{\pi} = \frac{1}{\gamma_{21}f_1^N + \gamma_{12}f_2^N} \begin{bmatrix} \gamma_{21}f_1^N & \gamma_{12}f_2^N \end{bmatrix}. \tag{4.54}$$

Note that the ratio of the stationary probabilities of the two cells is

$$\frac{\pi_1}{\pi_2} = \frac{\gamma_{21}f_1^N}{\gamma_{12}f_2^N} = \frac{(1-p)f_1^N}{pf_2^N}. \tag{4.55}$$

Since this ratio is not equal to

$$\left(\frac{f_1}{f_2}\right)^N, \tag{4.56}$$

the SEGS does not behave rationally except for $p = 0.5$. Moreover, note that for $p = 0.5$, the matrix of generation probabilities (4.52) is symmetric. Hence, this example illustrates that asymmetry of the matrix of generation probabilities may lead to behavior that is not rational. As a result, it is possible that, for small N, the most fit cell is not the most probable cell at steady state. For instance, take $p = 0.1$,

$N = 1$, $f_1 = 1$ and $f_2 = 2$. We obtain

$$\pi_1 = \frac{0.9}{0.9 + 0.2} \approx 0.818, \tag{4.57}$$

and

$$\pi_2 = \frac{0.2}{0.9 + 0.2} \approx 0.182. \tag{4.58}$$

This is why rationality is desired for behavior design and optimization, to ensure that fitter cells are more probable at steady state.

4.3.2 Symmetric Matrix of Generation Probabilities

Consider the evolutionary generation system (X, R, P, G), where

- $X = \{x_1, x_2\}$,
- $R = \{r_1, r_2\}$,
- $P(r_1) = p$, $P(r_2) = 1 - p$, $p \neq 0$,
- $G(x_1, r_1) = x_2$, $G(x_1, r_2) = x_1$, $G(x_2, r_1) = x_1$, and $G(x_2, r_2) = x_2$ (see Figure 4.3).

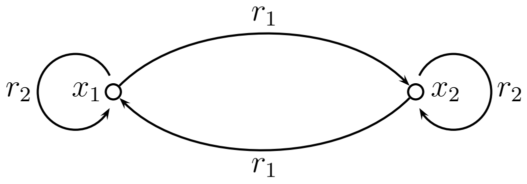


Figure 4.3: The directed graph of the example in Section 4.3.2.

The matrix of generation probabilities for this evolutionary generation system is

$$\boldsymbol{\gamma} = \begin{bmatrix} 1 - \gamma_{12} & \gamma_{12} \\ \gamma_{21} & 1 - \gamma_{21} \end{bmatrix} = \begin{bmatrix} 1 - p & p \\ p & 1 - p \end{bmatrix}. \tag{4.59}$$

Let $F(x_1) = f_1$ and $F(x_2) = f_2$. Let N be a finite level of selectivity. Utilizing (4.34), the matrix of transition probabilities for the selective evolutionary generation system is

$$\mathbf{P} = \frac{1}{f_1^N + f_2^N} \begin{bmatrix} f_1^N + (1-\gamma_{12}) f_2^N & \gamma_{12} f_2^N \\ \gamma_{21} f_1^N & (1-\gamma_{21}) f_1^N + f_2^N \end{bmatrix}. \tag{4.60}$$

This SEGS is both aperiodic and positive recurrent, and hence, ergodic.

The stationary distribution of the SEGS can be computed to be

$$\boldsymbol{\pi} = \frac{1}{\gamma_{21} f_1^N + \gamma_{12} f_2^N} \begin{bmatrix} \gamma_{21} f_1^N & \gamma_{12} f_2^N \end{bmatrix}. \tag{4.61}$$

The example has been constructed such that $\gamma_{12} = \gamma_{21} = p$, and so this ergodic SEGS is rational.

Taking partial derivatives, the extrinsic and intrinsic resilience equations of the two cells of the SEGS are

$$\rho_{11} = \frac{N \gamma_{21} f_1^{N-1}}{\gamma_{21} f_1^N + \gamma_{12} f_2^N} \left(1 - \frac{\gamma_{21} f_1^N}{\gamma_{21} f_1^N + \gamma_{12} f_2^N} \right), \tag{4.62}$$

$$= \frac{N \pi_1 (1-\pi_1)}{f_1} \neq 0, \tag{4.63}$$

$$\rho_{12} = \frac{-\gamma_{21} f_1^N}{(\gamma_{21} f_1^N + \gamma_{12} f_2^N)^2} N \gamma_{12} f_2^{N-1}, \tag{4.64}$$

$$= \frac{-N \pi_1 \pi_2}{f_2} \neq 0, \tag{4.65}$$

$$\rho_{21} = \frac{-\gamma_{12} f_2^N}{(\gamma_{21} f_1^N + \gamma_{12} f_2^N)^2} N \gamma_{21} f_1^{N-1}, \tag{4.66}$$

$$= \frac{-N \pi_2 \pi_1}{f_1} \neq 0, \tag{4.67}$$

$$\rho_{22} = \frac{N \gamma_{12} f_2^{N-1}}{\gamma_{21} f_1^N + \gamma_{12} f_2^N} \left(1 - \frac{\gamma_{12} f_2^N}{\gamma_{21} f_1^N + \gamma_{12} f_2^N} \right), \tag{4.68}$$

$$= \frac{N \pi_2 (1-\pi_2)}{f_2} \neq 0. \tag{4.69}$$

These equations match the theoretical results stated previously. If $\boldsymbol{\gamma}$ were not symmetric, this SEGS still happens to be resilient.

If $f_1 > f_2$, the mean hitting time to x_1 is

$$\boldsymbol{\sigma}_1 = \begin{bmatrix} 0 \\ \frac{f_1^N + f_2^N}{\gamma_{21} f_1^N} \end{bmatrix}, \tag{4.70}$$

and if $f_2 > f_1$, the mean hitting time to x_2 is

$$\boldsymbol{\sigma}_2 = \begin{bmatrix} \frac{f_1^N + f_2^N}{\gamma_{12} f_2^N} \\ 0 \end{bmatrix}. \tag{4.71}$$

In the limit as N approaches ∞, we have

$$\lim_{N\to\infty} \boldsymbol{\sigma}_1 = \begin{bmatrix} 0 \\ \frac{1}{\gamma_{21}} \end{bmatrix}, \tag{4.72}$$

and

$$\lim_{N\to\infty} \boldsymbol{\sigma}_2 = \begin{bmatrix} \frac{1}{\gamma_{12}} \\ 0 \end{bmatrix}, \tag{4.73}$$

as expected from the proof of Theorem 4.13.

4.4 Summary

Initiating the process of selective evolution requires answering the three basic questions:

1. What are the cells and resources?
2. How are offspring cells generated?
3. How is cell fitness measured?

Once these questions are answered, the SEGS algorithm proceeds according to (3.3) and produces a stationary distribution of cells that is rational, resilient and opportunistic. Model-based knowledge is not utilized by the algorithm.

The SEGS algorithm has many biological parallels. The answers to the above questions are also well known: during asexual reproduction, an additional cell is generated through a division of genetic material in a process called mitosis [105]. Each cell's genetic identity is parameterized by a sequence of symbols from a four letter alphabet, its *genotype*, while the physical realization of a genotype is known as its *phenotype*. Although reproduction occurs through operations on the genotype, it is the functionality of the phenotype that determines cell fitness [105]. Hence, it is necessary to account for the *genotype-phenotype mapping* when evaluating cell fitness in a SEGS.

4.5 Relationship Between Selective Evolutionary Generation Systems and Markov Chain Monte Carlo Algorithms

The SEGS algorithm is an example of a Markov Chain Monte Carlo (MCMC) algorithm. MCMC algorithms are useful for simulating large random fields through sampling, and are frequently employed in statistical mechanics applications [114]. MCMC algorithms utilize an irreducible, aperiodic, time-homogeneous Markov chain such that the stationary distribution, $\boldsymbol{\pi}$, is the target distribution. Since convergence to the target distribution is easier to check for reversible Markov chains, these Markov chains are the most frequent case of MCMC algorithms [114].

Hence, the design of an MCMC algorithm involves finding an ergodic transition matrix $\mathbf{P}$ that satisfies

$$\pi_i P_{ij} = \pi_j P_{ji}, \ \forall i, j. \tag{4.74}$$

According to [114], a typical choice of P_{ij} has the form

$$P_{ij} = Q_{ij}\alpha_{ij},\ \forall j \neq i. \tag{4.75}$$

Here, $\mathbf{Q}$ is a probability transition matrix (called the *candidate-generating matrix*) with elements Q_{ij} representing the probability of "tentatively" choosing a transition from i to j, and $\boldsymbol{\alpha}$ is a probability transition matrix with elements α_{ij} representing the probability of accepting that transition. A generic formulation for the acceptance probabilities is specified by the Hastings algorithm, which sets

$$\alpha_{ij} = \frac{s_{ij}}{1 + \frac{\pi_i Q_{ij}}{\pi_j Q_{ji}}}, \tag{4.76}$$

where s_{ij} are the elements of a symmetric matrix $\mathbf{S}$. Special cases of the Hastings algorithm include the Metropolis algorithm, which is used in simulated annealing, and Barker's algorithm.

Barker's sampler [139], from which Barker's algorithm takes its name, was proposed to compute radial distribution functions for plasmas over a wide range of temperatures and densities so that macroscopic thermodynamic variables could be calculated. The goal is to produce a Markov chain that, at steady state, results in a frequency distribution whereby a particle configuration state i recurs with a frequency proportional to the Boltzmann factor $\exp(-V_i/kT)$ for that state, where V_i is the potential energy of state i, k is the Boltzmann constant, and T is temperature. The acceptance probability for Barker's sampler is taken to be

$$\alpha_{ij}(T) = \frac{1}{1 + e^{-(V_i - V_j)/kT}}. \tag{4.77}$$

The acceptance probability for the more general Barker's algorithm sets $s_{ij} = 1$ in

(4.76), so that

$$\alpha_{ij} = \frac{1}{1 + \left(\frac{\pi_i}{\pi_j}\right)\left(\frac{Q_{ij}}{Q_{ji}}\right)}. \tag{4.78}$$

In the case of purely random $\mathbf{Q}$, this becomes

$$\alpha_{ij} = \frac{1}{1 + \left(\frac{\pi_i}{\pi_j}\right)}. \tag{4.79}$$

A SEGS has $\mathbf{Q} = \boldsymbol{\gamma}$. For rational behavior, we impose a symmetry condition so that $Q_{ij} = Q_{ji}$. Setting $s_{ij} = 1$ in (4.76), the definition of rational behavior implies that the acceptance probability utilized by the SEGS algorithm is

$$\alpha_{ij} = \frac{1}{1 + \left(\frac{\pi_i}{\pi_j}\right)}. \tag{4.80}$$

Thus, the SEGS algorithm and Barker's algorithm are the same.

However, this dissertation arrived at Barker's algorithm in a non-traditional manner, i.e., we did not assume time-reversibility and begin at Hasting's algorithm. Instead, modeling Nature, we started with a self-reproducing process and selected according to local rationality. The aim was to achieve global rational behavior, thereby resulting in resilience and opportunism. A required assumption was equiprobable forward and reverse transitions prior to selection, a fact borne out in Nature. This assumption resulted in the SEGS algorithm being time-reversible. Furthermore, efficient searching suggested exponential fitness functions, which are also documented in Nature. The combination of an exponential fitness function and a SEGS algorithm is the Barker sampler.

In light of the connection between rational behavior and statistical mechanics through entropy (see Table 4.1), it is perhaps unsurprising that selective evolutionary generation results in Barker's version of an MCMC method.

4.6 Multi-Selective Generation Systems as Markov Chains That Behave Rationally

Unfortunately, a SEGS may require a large number of steps to generate a candidate whose fitness exceeds a threshold. Therefore, one goal of this section is to extend the work of this chapter so that resilient and opportunistic behavior design is inexpensively attained in fewer generations, on average. The increased speed is particularly important for the finite-horizon problem, when a fit candidate must be found within a pre-specified number of algorithm iterations.

Another goal of this section is to analyze the resilience of the behavior designed by the Canonical Genetic Algorithm with Fitness Proportional Selection (CGAFPS), through its use of rational behavior. Recall that the primary benefit of employing rational behavior is its capacity for optimal search, where optimality is defined as either a minimization of prior information or a maximization of search effort savings. A secondary benefit is that rational behavior, when applied to Markov chains (see Section 4.1.3), is a sufficient condition for resilience and opportunism. To facilitate the analysis, we develop the requisite extension of a SEGS so that the technique in this section is very similar to the CGAFPS. We show that the conditions for the extended technique to achieve rational behavior are highly restrictive, and that there are instances when a SEGS technique finds fit candidates faster than the extended technique.

The implication is that the SEGS scheme, which also employs rational behavior, is more generally applicable for optimal search and resilient behavior design than the CGAFPS. Hence, the CGAFPS must be modified for use in optimization with dynamic fitness landscapes if rational behavior is desired. However, we also show that the CGAFPS typically finds fit candidates faster than a SEGS for non-resilient behavior design; this trade-off is consistent with the No Free Lunch theorem for optimization [140].

This section generalizes selective evolutionary generation by describing the concept of multi-selective generation, which utilizes the fitness value of a candidate optimizer, all fitness values in a neighborhood of the candidate optimizer, and the level of selectivity. That is, multi-selective generation assumes a more global knowledge of the reward function than selective generation, but the trade-off is faster behavior design.

More specifically, the following subsections utilize a novel, extended mathematical definition of selection, the *Multi-Select* function, for use in behavior design. This function is used to prove that multi-selective generation is a sufficient condition for rational behavior, under certain technical assumptions. Since rational behavior is itself a sufficient condition for resilience and opportunism, the resultant multi-selective generation scheme is therefore resilient and opportunistic. The limitations imposed by the technical assumptions are then discussed, along with their relevance to the CGAFPS. Lastly, the convergence properties of multi-selective generation are compared to a SEGS.

The authors of [28] (a recent overview of evolutionary computation for dynamic fitness landscapes) state that ‘there are no published results that are comparative to the patentable works cited for static environments,’ a failing that this section seeks to remedy.

4.6.1 Theoretical Extensions

The following definitions are required to extend the theoretical foundations of selective evolutionary generation to multi-selective generation.

Definition 4.8. A *generation system* is a triple $\mathcal{E} = (X, R, G)$, where

- X is a *set of* n *cells*, $X = \{x_1, x_2, \ldots, x_n\}$;
- R is a *set of* m *resources*, $R = \{r_1, r_2, \ldots, r_m\}$, that can be utilized for cell reproduction;

- $G : X \times R \to X$ is a *generation function* that maps a parent cell and a resource into a descendant cell outcome.

Definition 4.9. The set of cells, X, of the generation system $\mathcal{E} = (X, R, G)$ is *reachable* through G and R if, for all pairs $x_1,\ x_2 \in X$, there exists $k \in \mathbb{N}$ and a sequence $(r_k) \in R$ such that $x_2 = G\left(x_1, (r_k)\right)$.

Definition 4.10. Given a cell set, X, and a fitness function $F : X \to \mathbb{R}^+$, let *Multi-Select* : $X^k \times \mathbb{N} \to X$ be a random function such that if $x_1, \ldots, x_k \in X^k$ are any k cells, and $N \in \mathbb{N}$ is the *level of selectivity*, then

$$\textit{Multi-Select}(x_1, \ldots, x_k, N) = x_i,\ \text{with probability}\ \frac{F\left(x_i\right)^N}{\sum\limits_{j=1}^{k} F\left(x_j\right)^N},\ \ 1 \le i \le k. \tag{4.81}$$

We can now define a multi-selective generation system (MSGS).

Definition 4.11. A *multi-selective generation system* is a quadruple $\Gamma = (X, R, G, F)$, where

- (X, R, G) is a generation system;
- $F : X \to \mathbb{R}^+$ is a function that evaluates cell fitness;
- the set of cells, X, is reachable through G and R; and
- the dynamics of the system are given by

$$\mathcal{X}(t+1) = \textit{Multi-Select}\left(\mathcal{X}(t), G\left(\mathcal{X}(t), r_1\right), \ldots, G\left(\mathcal{X}(t), r_m\right), N\right). \tag{4.82}$$

In (4.82), $\mathcal{X}(t)$ denotes the realization of a random cell variable at time t, r_i is a resource where $1 \le i \le m$, $G(\mathcal{X}(t), r_i)$ denotes the offspring of the realized random cell utilizing resource r_i at time t, and $\mathcal{X}(0)$ has a known probability mass function. The cells $G\left(\mathcal{X}(t), r_i\right),\ 1 \le i \le m$, constitute the largest neighborhood of $\mathcal{X}(t)$ within which a transition is possible. The fitness values of cells in this neighborhood are required at each step.

Also in (4.82), the probability of a cell realization at some future time given the present cell realization is conditionally independent of the past time history of cell realizations. Thus, the dynamics of an MSGS form a discrete-time homogeneous Markov chain. This property is useful for the MSGS analysis that follows.

Like the *Select* function, the *Multi-Select* function has a number of interesting properties, including:

- For all N and for all $1 \leq i \leq k,\ 1 \leq j \leq k$,

$$\frac{\Pr[\mathit{Multi\text{-}Select}(x_1, \ldots, x_k, N) = x_i]}{\Pr[\mathit{Multi\text{-}Select}(x_1, \ldots, x_k, N) = x_j]} = \left(\frac{F(x_i)}{F(x_j)}\right)^N. \tag{4.83}$$

  That is, the ratio of the probabilities of selecting any two cells is equal to the ratio of their respective fitnesses raised to the power N.

- For $N = 0$, the values of fitnesses are irrelevant. That is, for all $1 \leq i \leq k$,

$$\Pr[\mathit{Multi\text{-}Select}(x_1, \ldots, x_k, 0) = x_i] = 1/k. \tag{4.84}$$

- When $N \to \infty$, if there is a unique index, I, such that $F(x_i)$ is maximized for $i = I$ then

$$\Pr[\mathit{Multi\text{-}Select}(x_1, \ldots, x_k, N) = x_I] \to 1. \tag{4.85}$$

- If all the fitnesses are equal then, for all N and for all $1 \leq i \leq k$,

$$\Pr[\mathit{Multi\text{-}Select}(x_1, \ldots, x_k, N) = x_i] = 1/k. \tag{4.86}$$

Section 3.2 demonstrates that, for each iteration of the CGAFPS, the ratio of the probability of selecting an unchanged cell as a member of the population for the next generation to the probability of selecting an offspring of this cell (i.e., a mutated and/or recombined version of the cell) as a member of the population for the next generation is proportional to the fitness ratio of this cell and its offspring. If the constant of proportionality is one, then a particular case of (4.83) is obtained with $N = 1$. For this section, our extension of the SEGS scheme is such that there is another similarity with the CGAFPS: fitness proportional selection is a particular case of multi-selective generation with $N = 1$.

The concept of multi-selective generation has been previously implemented experimentally with great success. Consider the well-known paper, [141], which describes a system for the evolution of virtual creatures in a fitness landscape that changes frequently because of competition. The work utilizes an *all vs. best* strategy, defined as the competition between all individuals in a generation and a single opponent with the highest fitness from the previous generation. This strategy is what we have called multi-selective generation. The paper states that

> 'the most "interesting" results occurred when the all vs. best competition pattern was used. Both one and two species evolutions produced some intriguing strategies.'

To apply the theory of rational behavior for time-homogeneous, irreducible, ergodic Markov chains, we need the following.

Definition 4.12. Let $\Gamma = (X, R, G, F)$ be a multi-selective generation system. Let $x_i, x_j \in X$ be any two cells. The *descendancy matrix*, $\boldsymbol{\delta}$, has elements

$$\delta_{ij} = \begin{cases} 1 & \text{if } \exists r \in R : x_j = G(x_i, r),\ 1 \le i, j \le n, \\ 0 & \text{otherwise.} \end{cases} \tag{4.87}$$

Hence, the descendancy matrix indicates whether it is possible to produce cell x_j in one step from cell x_i, using any resource. Recall that an MSGS follows the stochastic

Markov process described by (4.82). The descendancy matrix is used when specifying a matrix of transition probabilities that describes the cell-to-cell transitions that occur as a result of the multi-selection dynamics. For the MSGS $\Gamma = (X, R, G, F)$, the *matrix of transition probabilities*, $\mathbf{P}$, has elements

$$\begin{aligned}
P_{ij} =& \Pr[\mathcal{X}(t+1) = x_j \mid \mathcal{X}(t) = x_i], && (4.88)\\
=& \Pr[\mathit{Multi\text{-}Select}\left(x_i, G\left(x_i, r_1\right), \ldots, G\left(x_i, r_m\right), N\right) = x_j \mid \mathcal{X}(t) = x_i] \times \\
& \Pr[\text{offspring is } x_j \mid \text{progenitor is } x_i] && (4.89)\\
=& \begin{cases} \frac{F(x_j)^N}{\sum\limits_{k=1}^{m} F(G(x_i,r_k))^N + F(x_i)^N} \delta_{ij}, & \forall j \neq i, \\ 1 - \sum\limits_{\substack{j=1\\ j\neq i}}^{n} \frac{F(x_j)^N}{\sum\limits_{k=1}^{m} F(G(x_i,r_k))^N + F(x_i)^N} \delta_{ij}, & \text{if } j = i. \end{cases} && (4.90)
\end{aligned}$$

4.6.2 Dynamic Properties of Multi-Selective Generation Systems

We can now state some dynamic properties of multi-selective generation systems, under certain technical conditions.

Theorem 4.14. *For the ergodic MSGS $\Gamma = (X, R, G, F)$, assume that*

i) the descendancy matrix, $\boldsymbol{\delta}$, is symmetric, and

ii) $\forall 1 \leq i, j \leq n$ with $\delta_{ij} = 1$,

$$\sum_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N = \sum_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N. \tag{4.91}$$

Then the Markov chain representing the stochastic dynamics of the ergodic MSGS

1. *behaves rationally with fitness F and level N. That is, the row vector $\boldsymbol{\pi} = \begin{bmatrix} \pi_1 & \pi_2 & \ldots & \pi_n \end{bmatrix}$, where π_i satisfies (4.2), is a left eigenvector of $\mathbf{P}$, the matrix*

of transition probabilities for Γ, with corresponding eigenvalue 1 (i.e., $\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}$). Hence, $\boldsymbol{\pi}$ is the vector of stationary probabilities for the MSGS.

2. *is time-reversible, i.e.,*

$$\pi_i P_{ij} = \pi_j P_{ji}, \ \forall i, j. \tag{4.92}$$

Proof. See Appendix A. □

As a result of Theorem 4.6, the stochastic dynamics of the ergodic MSGS with sufficient conditions i) and ii) are resilient and opportunistic.

The symmetry condition i) on the descendancy matrix, $\boldsymbol{\delta}$, implies that there exists a forward and reverse transition between any pair of cells. This condition is similar to the one in Section 4.2.2.

Condition ii) is a restrictive sufficient condition. It states that the sum of the fitness values of possible transitions in the neighborhood of x_i, which includes x_j, is equal to the sum of the fitness values of possible transitions in the neighborhood of x_j, which includes x_i. Although this condition is an extension of one implicitly assumed in Section 4.2.2, which is that the addition of any two fitness values commute, the extended condition ii) may be difficult to satisfy.

If condition ii) is satisfied, then there is a need to evaluate the fitness of cells in a sub-population of candidate optimizers (as in the CGAFPS). For $m > 2$, an ergodic MSGS is more computationally expensive than an ergodic SEGS, but can be less expensive than evaluating the fitness of all elements in the domain of the objective function at the start of the search.

Necessary conditions for rational behavior are as follows.

Theorem 4.15. *For the ergodic MSGS $\Gamma = (X, R, G, F)$, assume that the Markov chain representing the stochastic dynamics of the ergodic MSGS behaves rationally* *with fitness F and level N. Then*

$$\sum_{i=1}^{n} \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} \delta_{ji} = \sum_{i=1}^{n} \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij}. \tag{4.93}$$

If the Markov chain representing the stochastic dynamics of the ergodic MSGS is also time-reversible, then

$$\frac{\delta_{ji}}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} = \frac{\delta_{ij}}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N}. \tag{4.94}$$

Proof. See Appendix A. ☐

For finite N and cell fitness values, (4.94) is thus a necessary and sufficient condition for rational behavior. Therefore, not satisfying (4.94) results in behavior that is not rational. Since the CGAFPS is not often applied such that (4.94) is true with $N = 1$, the CGAFPS does not achieve rational behavior in these instances. Modification is required.

4.6.3 Convergence Properties of Multi-Selective Generation Systems

We now compare the convergence rates of an ergodic MSGS and an ergodic SEGS, by utilizing the second largest eigenvalue as a measure of convergence rate — the smaller this value, the more quickly the Markov chain dynamics converge to steady state. We make use of the result in [142] for reversible Markov chains with a common underlying graph. Let the underlying graph representations of the ergodic MSGS and the ergodic SEGS be the same, and let the sufficient conditions for rational behavior with fitness F with level N be satisfied. If $\lambda_2^M < 1$ is the second largest eigenvalue of the MSGS probability transition matrix $\mathbf{P}^M$, and $\lambda_2^S < 1$ is the second largest eigenvalue of the SEGS probability transition matrix $\mathbf{P}^S$ then, from [142],

$$\left(1-\lambda_2^M\right) \geq \frac{\min\limits_{\{i,j\} \text{ is an edge}} \frac{w_{ij}^M}{w_{ij}^S}}{\max\limits_{i} \frac{\pi_i^M}{\pi_i^S}} \left(1-\lambda_2^S\right), \tag{4.95}$$

where

$$w_{ij}^M = \pi_i^M P_{ij}^M, \tag{4.96}$$

$$w_{ij}^S = \pi_i^S P_{ij}^S. \tag{4.97}$$

Because both techniques yield rational behavior, $\pi_i^M = \pi_i^S$. Consequently,

$$\left(1-\lambda_2^M\right) \geq \left(\min_{\{i,j\} \text{ is an edge}} \frac{P_{ij}^M}{P_{ij}^S}\right) \left(1-\lambda_2^S\right), \tag{4.98}$$

or

$$\left(1-\lambda_2^M\right) \geq \alpha \left(1-\lambda_2^S\right), \tag{4.99}$$

where, from (4.34) and (4.90),

$$\alpha = \min_{\substack{\{i,j\} \\ \text{is an edge}}} \frac{F(x_i)^N + F(x_j)^N}{\gamma_{ij} \left(\sum\limits_{\substack{k=1 \\ G(x_i,r_k) \neq x_j}}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N + F\left(x_j\right)^N \right)}, \tag{4.100}$$

where γ_{ij} is a probability value on the interval $(0,1]$.

In (4.99), if $\alpha = 1$, then $\lambda_2^M \leq \lambda_2^S$. Since $N \rightarrow \infty$ implies that $\alpha \rightarrow 1$, non-resilient multi-selective generation converges to steady state faster than a SEGS process.

In (4.99), if $\alpha > 1$, then $\lambda_2^M \leq \lambda_2^S$. This occurs for a typical application of the SEGS, where the number of resources is large enough so that the probability distribution on these resources (which is indicated by γ_{ij}, and is typically a uniform probability distribution) is small, yet the number of resources is also small enough so that the corresponding MSGS implementation does not select among large fitness neighborhoods at each generation. For this scenario, a SEGS requires a lot more exploration than the MSGS.

In (4.99), if $0 < \alpha < 1$, then $\lambda_2^M \leq \alpha\lambda_2^S + 1 - \alpha$. That is, for small α, it is possible that $\lambda_2^S \leq \lambda_2^M$. This occurs when γ_{ij} is large and N is small. The physical interpretation of this scenario is as follows: the resource choice (indicated by γ_{ij}) is biased in such a way that the probability of SEGS transitions to cells of higher fitness is greater than the probability of corresponding MSGS transitions. Hence, the faster SEGS convergence to steady state for this scenario. We believe that such a bias is possible, but is atypical.

The above leads to the following conclusion: while the CGAFPS may not satisfy the necessary conditions for rationality and hence not design rational behavior, the algorithm will, in general, converge to steady state faster than a SEGS. Such a trade-off is consistent with the No Free Lunch theorem for optimization [140].

4.6.4 Illustrative Example of a Multi-Selective Generation System

For simplicity, we illustrate the theory in this section with an example that is easy to analytically compute.

Consider the generation system (X, R, G), where $X = \{x_1, x_2, x_3\}$, $R = \{r_1, r_2\}$, $G(x_1, r_1) = x_2$, $G(x_1, r_2) = x_3$, $G(x_2, r_1) = x_3$, $G(x_2, r_2) = x_1$, $G(x_3, r_1) = x_1$, and $G(x_3, r_2) = x_2$ (see Figure 4.4).

The descendancy matrix for this generation system is

$$\boldsymbol{\delta} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}. \tag{4.101}$$

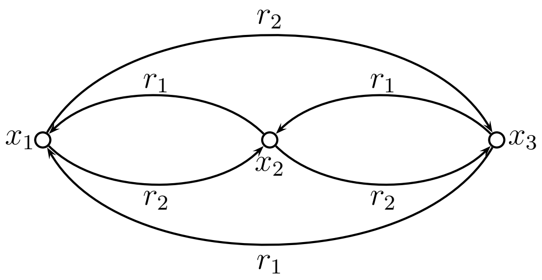


Figure 4.4: The directed graph of the example in Section 4.6.4.

Let $F(x_1) = f_1$, $F(x_2) = f_2$ and $F(x_3) = f_3$. Let N be a finite level of selectivity. Utilizing (4.90), the matrix of transition probabilities for the selective evolutionary generation system is

$$\mathbf{P} = \frac{1}{f_1^N + f_2^N + f_3^N} \begin{bmatrix} f_1^N & f_2^N & f_3^N \\ f_1^N & f_2^N & f_3^N \\ f_1^N & f_2^N & f_3^N \end{bmatrix}. \tag{4.102}$$

We see that the MSGS is both aperiodic and positive recurrent, and hence, ergodic.

The stationary distribution of the SEGS can be computed to be

$$\boldsymbol{\pi} = \frac{1}{f_1^N + f_2^N + f_3^N} \begin{bmatrix} f_1^N & f_2^N & f_3^N \end{bmatrix}. \tag{4.103}$$

The example has been constructed such that the sufficient conditions for rational behavior of the ergodic MSGS are satisfied.

Taking partial derivatives, the extrinsic and intrinsic resilience equations of the three cells of the MSGS satisfy (4.23) and (4.24). Since none of these equations are zero, the MSGS is resilient and opportunistic. Indeed, because of the cardinality of X and the feasible transitions specified by G, multi-selective generation is able to design, in one step, the desired behavior. This corresponds to the second largest eigenvalue of $\mathbf{P}$ being 0. Such speed is not matched by a SEGS, because the technique is limited to exploring one transition at a time.

4.6.5 Summary

Multi-selective generation extends a viable SEGS technique that uses rationality to achieve resilient and opportunistic behavior. Rational behavior is desirable because of its capacity for optimal search. However, the conditions for this extended scheme to behave rationally are highly restrictive. Since the technique is a generalization of the Canonical Genetic Algorithm with Fitness Proportional Selection (CGAFPS), it is unlikely that a typical application of the CGAFPS behaves rationally. Multi-selective generation can find a fit candidate optimizer faster than the original technique that it extends, but exceptions do exist. Therefore, the original SEGS scheme should be preferred for optimal search and resilient behavior design.

CHAPTER 5

EVOLUTION OF FLAPPING WING GAITS

5.1 Problem Description

One possible application of selective evolutionary generation systems is the on-line selection of flapping wing gaits during flight. This application requires resilient and opportunistic optimization because mission phase transitions may change the fitness of the current flapping gait. For instance, a micro air vehicle may scout a target by favoring a hovering form of flapping flight, engage the target after increasing the fitness of descending flapping gaits, and then quickly escape after deeming ascending gaits to be the most fit. Wind fluctuations within each mission phase are another example of possible fitness perturbations.

Current optimization of low Reynolds number flapping gaits requires multiple iterations of computationally expensive three dimensional flow simulations, on multiple nodes taking days, or even weeks, to complete [143]. Moreover, these simulations depend on flow model physics that are not well understood. Thus, there is a need for a computationally inexpensive, model-independent, resilient, opportunistic, global, and on-line selection technique for flapping wing flight.

There are examples of flapping gait evolution in the literature [144–147]. The results presented in these works are either complicated by hardware-specific interactions or derived from aerodynamic and hardware models with inaccurate assumptions (e.g., steady fluid flow) for simplicity. The tolerance to fitness function perturbations is also not examined. Our contributions in this area are unique because we achieve resilient and opportunistic flapping gaits without significant computation.

5.2 Surrogate Model

The following example applies the theory developed in this dissertation to a validated model that approximates the real-world physics of flapping flight. The model outputs a scalar for every acceptable input vector, and this scalar output makes it easy to discuss and verify claims of resilience and opportunism for a realistic application.

The surrogate model for hovering flight [148] predicts a lift coefficient, C_L, for a prescribed flapping motion with various input kinematic parameters. This flapping motion is described by

$$h(t) = h_a(t)\sin(\omega t), \tag{5.1}$$

$$\alpha(t) = 90 - \alpha_a(t)\sin\left(\omega t + \phi_\alpha\left(t\right)\right), \tag{5.2}$$

where $h_a(t) \in [1, 2]$ and $\alpha_a(t) \in [45, 80]$ are the piecewise-constant amplitudes of flapping stroke height and pitch respectively, ω is a frequency that depends on h_a and a constant Reynolds number of 100, and $\phi_\alpha(t) \in [60, 120]$ is the piecewise-constant phase shift angle for flapping pitch. The flapping motion described in (5.1)–(5.2) leads to the computation of a lift coefficient, C_L, through the surrogate model. Hence, the hovering flapping flight problem: given a time history of the target lift coefficient, $C_{L_{des}}(t)$, determine suitable time-varying flapping wing kinematic parameters that meet the target.

We utilize the following evolutionary generation system, (X, R, P, G), which is formulated as a random walk over a discretized search space.

- The set of cells, X, is the set of ordered triples $(h_a(t), \alpha_a(t), \phi_\alpha(t))$, where

$$h_a(t) \in \{1, 1.1, 1.2, \ldots, 1.9, 2\}, \tag{5.3}$$

$$\alpha_a(t) \in \{45, 46, 47, \ldots, 79, 80\}, \tag{5.4}$$

$$\phi_\alpha(t) \in \{60, 61, 62, \ldots, 119, 120\}. \tag{5.5}$$

- The set of resources, R, is the set $\{r_1, r_2, r_3, r_4, r_5, r_6\}$, with $r_i = \mathbf{e}_i,\ 1 \leq i \leq 6$ (where $\mathbf{e}_i$ are the standard basis vectors for $\mathbb{R}^6$). This choice of resources facilitates the perturbation of one of the elements of a cell in either a positive or negative direction when an offspring is generated.

- The probability mass function on R, P, is the discrete uniform distribution. This choice of probability mass function ensures that the matrix of generation probabilities is symmetric.

- The generation function, G, applied to X as

$$G\left((h_a(t), \alpha_a(t), \phi_\alpha(t)), r_i\right),\ 1 \leq i \leq 6, \tag{5.6}$$

is the triple given by

$$\begin{cases} \begin{bmatrix} 0.1 & -0.1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} r_i + \begin{bmatrix} h_a(t) \\ \alpha_a(t) \\ \phi_\alpha(t) \end{bmatrix}, \\ \text{if } 1 < h_a(t) < 2,\ 45 < \alpha_a(t) < 80, \\ \quad 60 < \phi_\alpha(t) < 120, \\ \\ (h_a(t), \alpha_a(t), \phi_\alpha(t)),\ \text{otherwise.} \end{cases} \tag{5.7}$$

Since the objective is for $C_L(t)$ to track $C_{L_{des}}(t)$, we use the fitness function

$$F\left(h_a\left(t\right), \alpha_a\left(t\right), \phi_\alpha\left(t\right)\right) = \exp\left(-\left(K_f\left(C_{L_{des}}(t) - C_L(t)\right)\right)^2\right), \tag{5.8}$$

where

$$K_f = 10, \tag{5.9}$$

and

$$C_L(t) = C_L\left(h_a\left(t\right), \alpha_a\left(t\right), \phi_\alpha\left(t\right)\right) \tag{5.10}$$

is the output of the surrogate model. Note that the fitness function in (5.8) has the following properties.

- Akin to a membership function, the fitness function is normalized so that a fitness between 0 and 1 is achieved depending on how well the model output matches the desired output. A fitness of 1 represents a perfect output match, whereas a fitness of 0 signifies a poor match.

- The fitness function utilizes a gain parameter, K_f, which indicates how dissimilar the desired output and a high-fitness true output are tolerated to be. Larger gains indicate that the SEGS is more permissive of poor matches. The gain parameter is also related to the level of selectivity, N, because the latter is always used as an exponent of fitness. Hence, in the above fitness function, K_f plays a similar role to N.

- Corresponding to the above, it can be shown that the fitness function is proportional to a Gaussian probability density function with mean equal to the desired output, variance equal to $\frac{1}{2K_f^2}$, and a constant of proportionality equal to $\sqrt{\frac{2\pi}{K_f^2}}$.

5.3 Surrogate Model Results

A sample run of the evolution scheme when $N = 5$ is depicted in Figures 5.1 to 5.4. A cell triple that achieves satisfactory performance is found within 1000 generations, and the scheme is resilient because it quickly finds a new triple that achieves an acceptable output when the target lift coefficient, and hence the fitness function, changes. In Figures 5.1 to 5.3, the red vertical dashed lines indicate a generation for which the evolved flapping forward and backward motion is illustrated in Figure 5.4.

For generations 1, 900, 1025, and 2000, the plots in Figure 5.4 each display 10 snapshots of a 15% elliptical airfoil through a flapping half-stroke. The solid circle represents the leading edge of the airfoil, which moves in an aircraft body-fixed reference frame with neutral position at (0,0). The arrows on the forward half-stroke plots indicate that the airfoil travels from the most rearward position to the most forward position, whereas the opposite is true for a backward half-stroke. Although the periods of the strokes vary at different generations because of the constant Reynolds number, the snapshots are taken at the same fractional period interval. Therefore, a stroke with more spacing between snapshots has a faster motion than a stroke with snapshots that are closely spaced.

Typically, the scheme averages 1 minute 18 seconds to compute the output of 1000 generations while running in MATLAB on a 2.50 GHz dual-core processor laptop with 4.00 GB of RAM and the Windows Vista operating system.

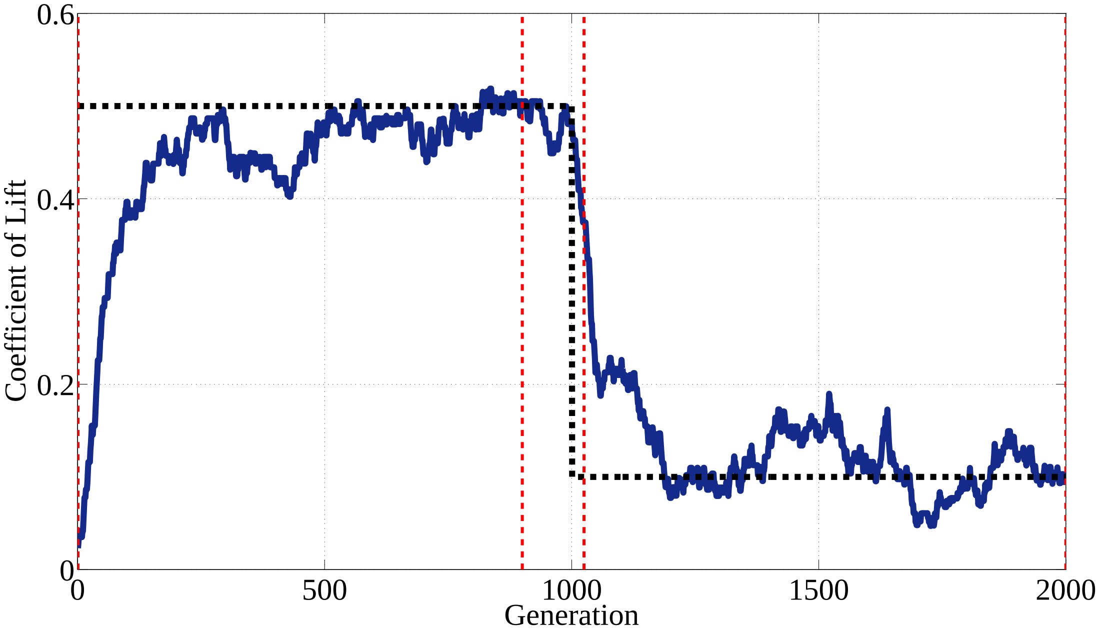


Figure 5.1: Target (dashed) and actual (solid) lift coefficients per generation.

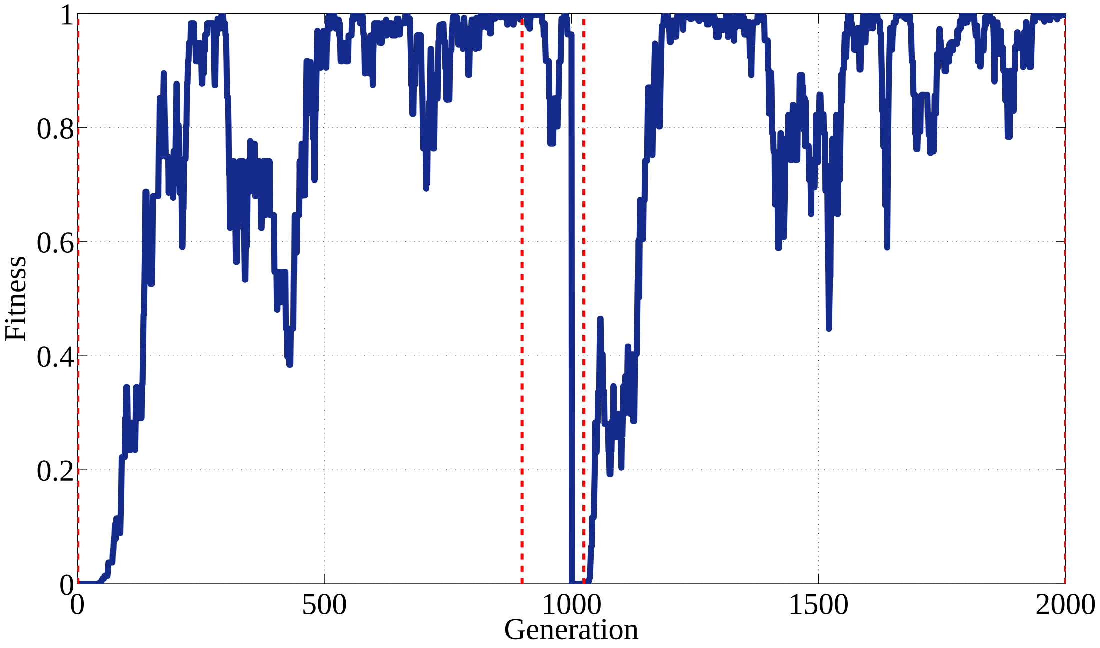


Figure 5.2: Fitness per generation.

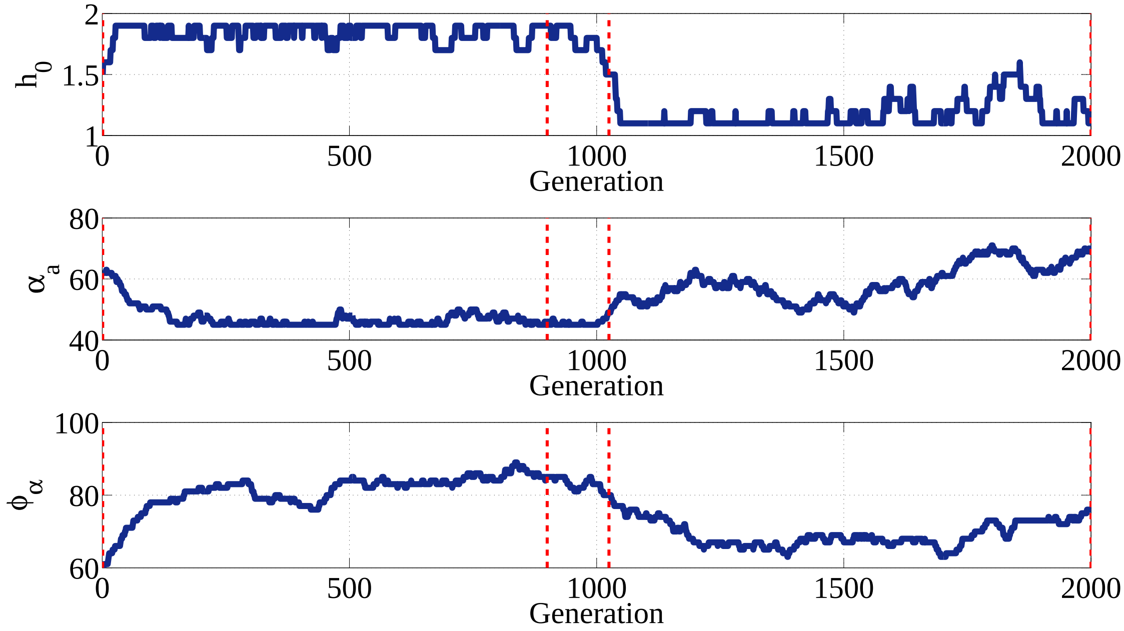
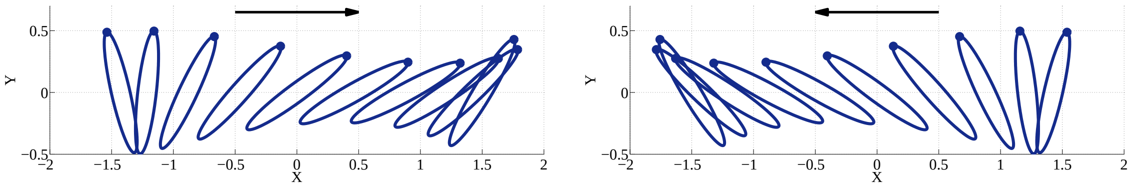
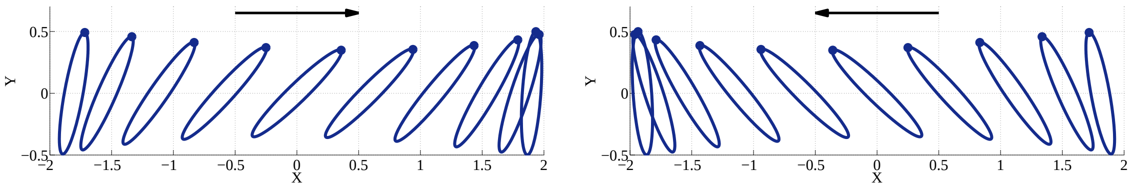


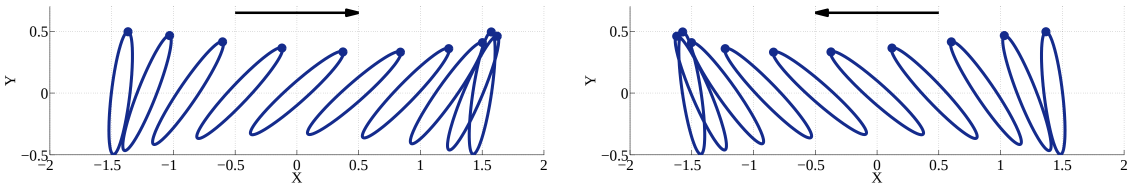
Figure 5.3: Flapping wing kinematic parameters per generation.



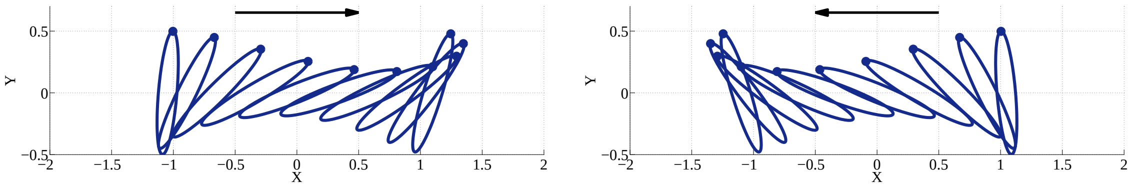
(a) Gen. 1: Forward Half-Stroke (b) Gen. 1: Back Half-Stroke



(c) Gen. 900: Forward Half-Stroke (d) Gen. 900: Back Half-Stroke



(e) Gen. 1025: Forward Half-Stroke (f) Gen. 1025: Back Half-Stroke



(g) Gen. 2000: Forward Half-Stroke (h) Gen. 2000: Back Half-Stroke

Figure 5.4: Snapshots of the forward and back half-strokes of the flapping wing sampled at the 1st, 900th, 1025th, and 2000th generations.

5.4 Surrogate Model Benchmarks

This section examines the effects of parameter variations of the SEGS detailed in Section 5.2. Comparisons to implementations of the canonical genetic algorithm with fitness proportional selection and the (1+1) evolutionary strategy are also included.

5.4.1 Level of Selectivity Effects

Sample runs of the SEGS for various levels of selectivity are plotted in Figures 5.5 to 5.7. For this comparison, the SEGS was initialized to the triple $(h_a(0), \alpha_a(0), \phi_\alpha(0))$ $= (1.5, 62, 60)$, and the target lift coefficient was held constant at $C_L = 0.5$. These figures illustrate the rationale for choosing $N = 5$ in Section 5.3. At low levels of selectivity ($N = 0.5$ and $N = 1$), the SEGS wanders through the search space and does not reach the target lift coefficient within a user-specified limit of 1000 generations. Increases in the level of selectivity cause a corresponding improvement in target lift coefficient tracking. The $N = 5$ trajectory depicts excursions away from the desired lift coefficient; these excursions are minimized at the slightly higher level of selectivity, $N = 10$. The $N = 100$ trajectory achieves near perfect lift coefficient tracking with few excursions. A suitable choice of the level of selectivity that tolerates excursions is therefore either $N = 5$ or $N = 10$, since excursions are one indicator of resilience. Another indicator of resilience is the initial behavior of the $N = 5$ and $N = 10$ trajectories; however, these two trajectories are approximately equal during the first 50 generations. Since the $N = 5$ trajectory achieves tracking and greater resilience than the $N = 10$ trajectory, we choose the level of selectivity $N = 5$.

The trade-off between optimality and resilience is documented in Figures 5.8 through 5.11. Figures 5.8 and 5.9 show simulations where the target lift coefficient varies frequently during 1000 SEGS generations, and the target includes a $C_L = 0.7$ value that is beyond the flapping wing capabilities that the surrogate model simulates.

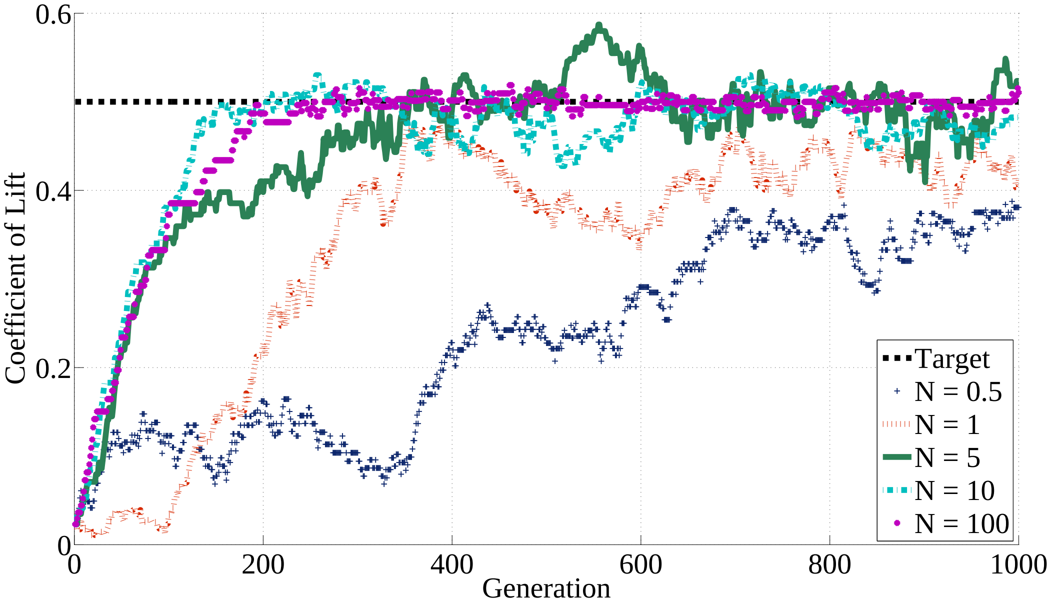


Figure 5.5: Level of selectivity effects on lift coefficients per generation.

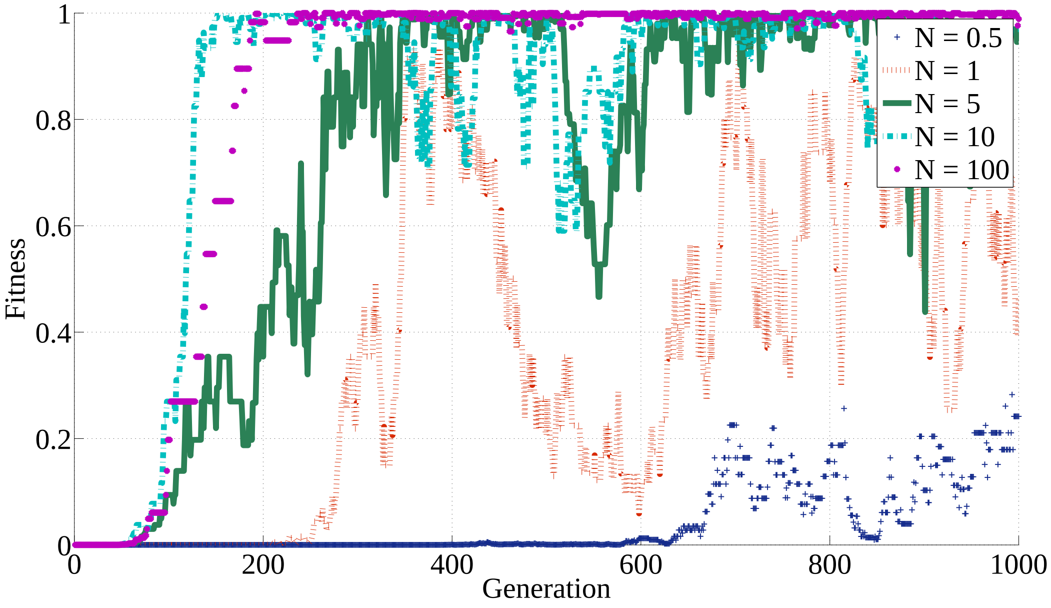


Figure 5.6: Level of selectivity effects on the fitness per generation.

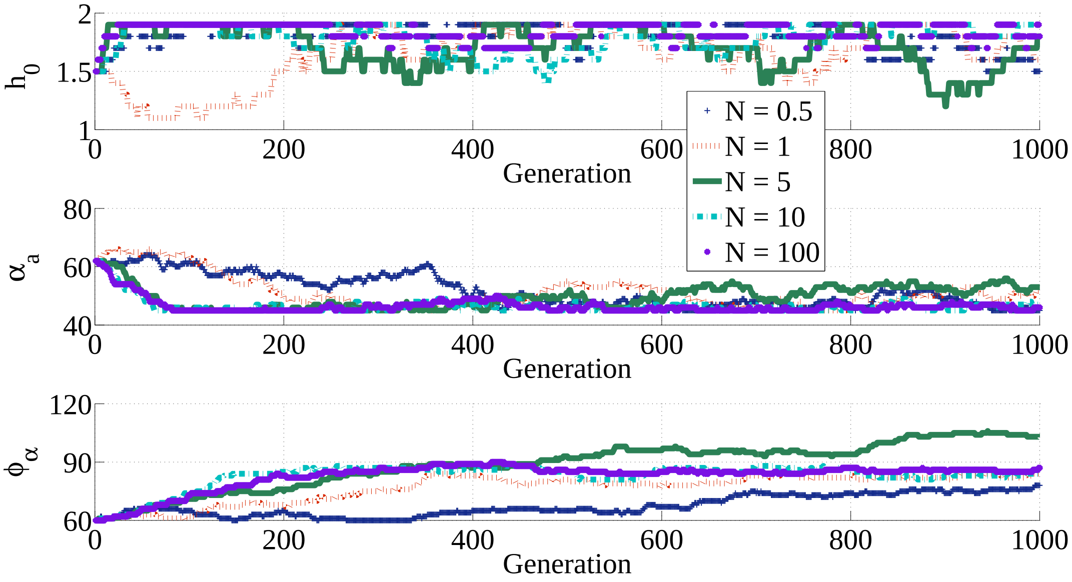


Figure 5.7: Level of selectivity effects on kinematic parameters per generation.

When the target lift coefficient variations are large, it is clear from Figure 5.8 that the $N = 5$ trajectory displays a more immediate response to the change in target than the $N = 100$ trajectory. This response is also evident in the initial higher fitness values of the $N = 5$ trajectory in Figure 5.9. However, the more selective $N = 100$ trajectory overtakes the $N = 5$ trajectory after a short period of time, in accordance with the shorter convergence times and optimality properties of high levels of selectivity. Both trajectories handle an unattainable target similarly. The resilience effects of a lower level of selectivity are not as pronounced for small target variations (see Figures 5.10 and 5.11).

The effect of the level of selectivity on one possible stopping criterion is outlined in Table 5.1. The table lists the average number of generations required to find a flapping wing gait with a lift coefficient that is within ±3% of the target value. This tolerance corresponds to a fitness value that is at least 0.975 or greater. As expected, an increase in N decreases the number of generations to find a "good" solution.

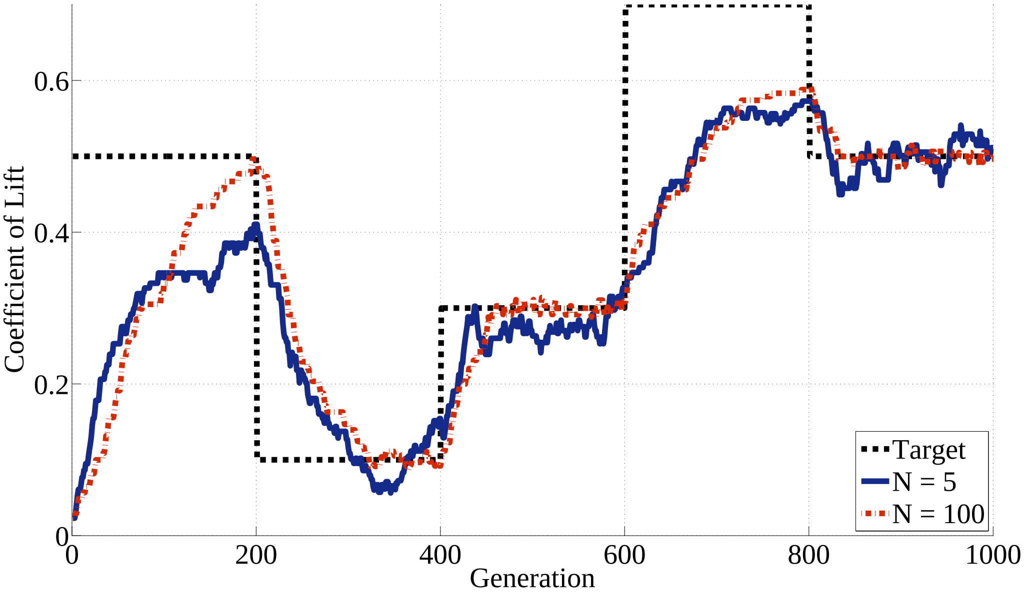


Figure 5.8: Target variations and actual lift coefficients per generation, illustrating the trade-off between optimality and resilience.

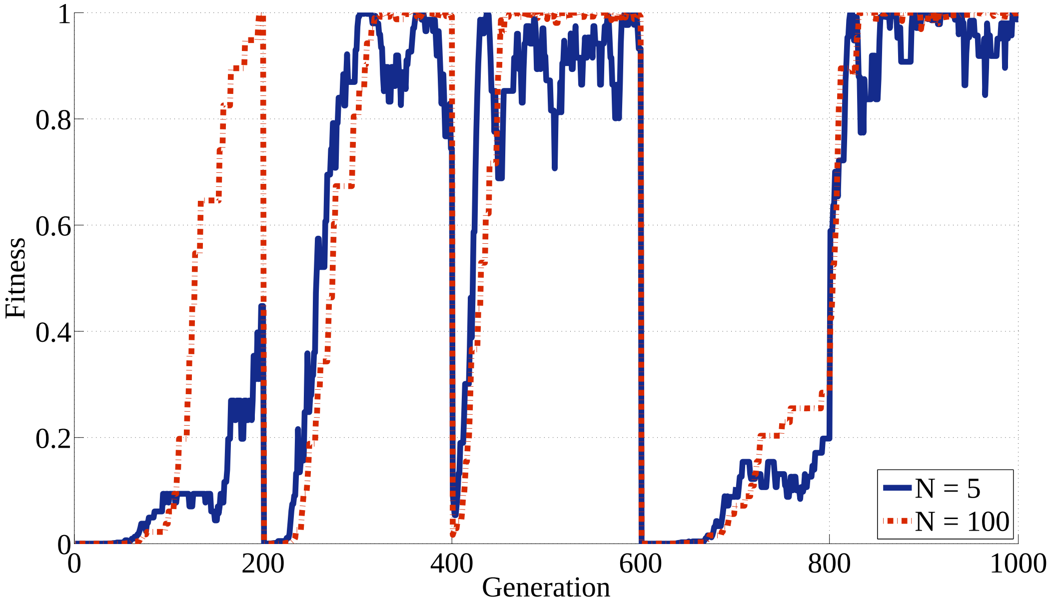


Figure 5.9: Fitness per generation, illustrating the trade-off between optimality and resilience.

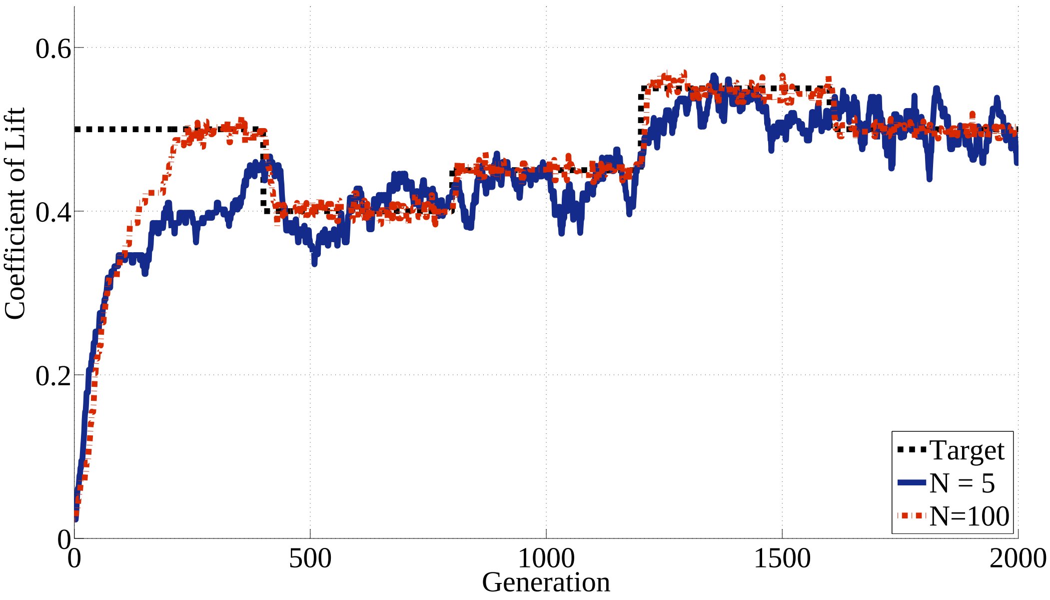


Figure 5.10: Target fluctuations and actual lift coefficients per generation.

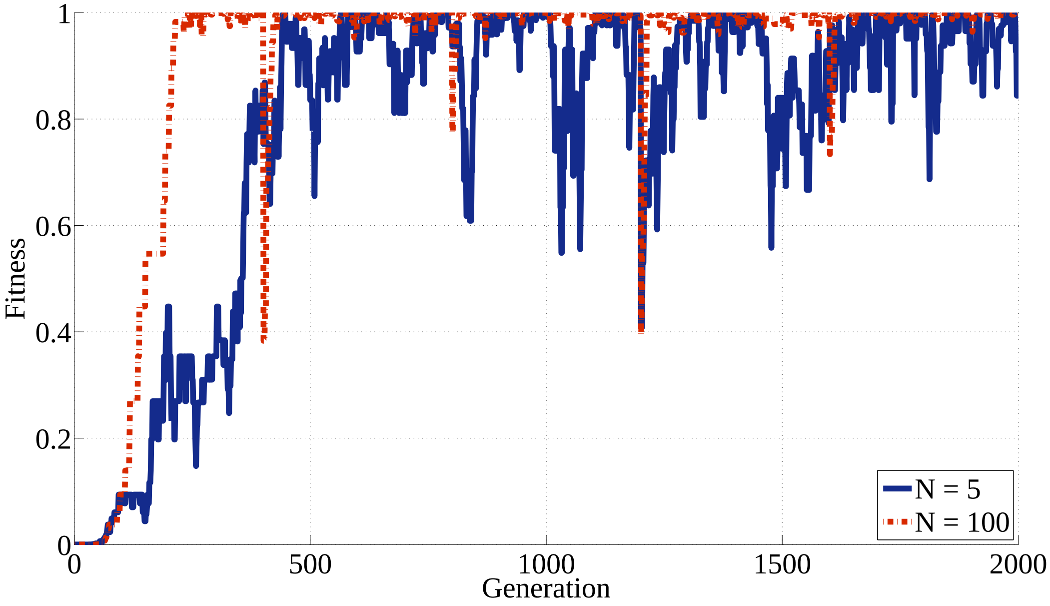


Figure 5.11: Fitness per generation for a fluctuating target.

Table 5.1: Level of Selectivity Effects on a Stopping Criterion

N	Initial Conditions	Target C_L	Average Number of Generations
1	(1.5,62,60)	0.5	1218
5	(1.5,62,60)	0.5	399
10	(1.5,62,60)	0.5	246
100	(1.5,62,60)	0.5	191

5.4.2 Initialization Effects

The initial conditions of a SEGS algorithm affect performance. Consider Table 5.2, which displays the effect of various initializations on the average number of generations required to find a flapping wing gait with a lift coefficient that is within ±3% of the target. As the table indicates, there is significant disparity in the average number of generations that is required before the stopping criterion is reached. Hence, initial conditions do play a role in the convergence of the SEGS algorithm. Moreover, the number of fit solutions in the search space also affect convergence. The surrogate model employed by the SEGS was trained on 24 samples of two dimensional computational fluid dynamics data, of which four samples had a lift coefficient of approximately 0.5 (the most number of samples for a given lift coefficient), and only one sample had a lift coefficient above 0.6 [143]. Accordingly, the effect of the initializations in Table 5.2 on the average number of generations required to find a flapping wing gait with lift coefficient within ±3% of 0.62 is tabulated in Table 5.3. It is clear that a significantly greater average number of generations is required when there are fewer fit solutions in the search space.

Table 5.2: Initialization Effects on a Stopping Criterion

N	Initial Conditions	Target C_L	Average Number of Generations
5	(1.5,62,60)	0.5	399
5	(1.5,45,60)	0.5	370
5	(1.5,62,90)	0.5	79
5	(1,45,60)	0.5	368
5	(1,45,120)	0.5	1
5	(1,80,60)	0.5	466
5	(1,80,120)	0.5	120
5	(2,45,60)	0.5	390
5	(2,45,120)	0.5	94
5	(2,80,60)	0.5	328
5	(2,80,120)	0.5	120

5.4.3 Discretization Effects

The SEGS in Section 5.2 discretizes the search space into h_a step sizes of 0.1, and into α_a and ϕ_α step sizes of 1 degree. The type of discretization employed by the SEGS affects the average number of generations required to find a flapping wing gait subject to the stopping criterion previously outlined. Table 5.4 provides the details for possible discretizations, with $N = 5$, initial conditions $(h_a(0), \alpha_a(0), \phi_\alpha(0)) = (1.5, 62, 60)$, and target lift coefficient 0.5.

The table hints at the prospect of an optimal discretization of the search space that minimizes the average number of generations required to find a fit flapping wing

Table 5.3: Initialization Effects on a Stopping Criterion with Few Fit Solutions

N	Initial Conditions	Target C_L	Average Number of Generations
5	(1.5,62,60)	0.62	1171
5	(1.5,45,60)	0.62	967
5	(1.5,62,90)	0.62	468
5	(1,45,60)	0.62	1065
5	(1,45,120)	0.62	80
5	(1,80,60)	0.62	1280
5	(1,80,120)	0.62	150
5	(2,45,60)	0.62	995
5	(2,45,120)	0.62	1
5	(2,80,60)	0.62	820
5	(2,80,120)	0.62	161

Table 5.4: Discretization Effects on a Stopping Criterion

Discretization Type	h_a Step	α_a Step	ϕ_α Step	Average Number of Generations
Fine	0.05	0.5	0.5	918
Baseline	0.1	1	1	399
Coarse	0.2	5	5	49
Very Coarse	0.5	10	10	58

gait. However, care must be taken to not use too coarse a discretization in the quest for reduced computation, since such a discretization may omit subtle features of the search space. It is expected that the optimal discretization be application dependent.

5.4.4 Comparison to Other Algorithms

Here, we compare the SEGS algorithm to implementations of the canonical genetic algorithm with fitness proportional selection and the (1+1) evolutionary strategy. These two algorithms are chosen because of their similarity to the SEGS approach. A description and analysis of these algorithms is contained in Section 3.2.

A sample run of the canonical genetic algorithm with fitness proportional selection is depicted in Figures 5.12 to 5.14. The algorithm was initialized with a population of four triples: $(1.5, 62, 60)$, $(1.0, 45, 60)$, $(2.0, 80, 120)$ and $(1.5, 62, 90)$, resulting in a higher starting fitness than the SEGS approach. A cell triple that achieves satisfactory performance is found within 1000 generations. Typically, the scheme averages 1 minute 47 seconds to compute the output of 1000 generations while running in MATLAB on a 2.50 GHz dual-core processor laptop with 4.00 GB of RAM and the Windows Vista operating system. This time is longer than the average computation time for the SEGS algorithm.

A sample run of the (1+1) evolutionary strategy is depicted in Figures 5.15 to 5.17. A cell triple that achieves satisfactory performance is found within 1000 generations. Typically, the scheme averages 1 minute 18 seconds to compute the output of 1000 generations while running in MATLAB on a 2.50 GHz dual-core processor laptop with 4.00 GB of RAM and the Windows Vista operating system. This time is identical to the average computation time for the SEGS algorithm.

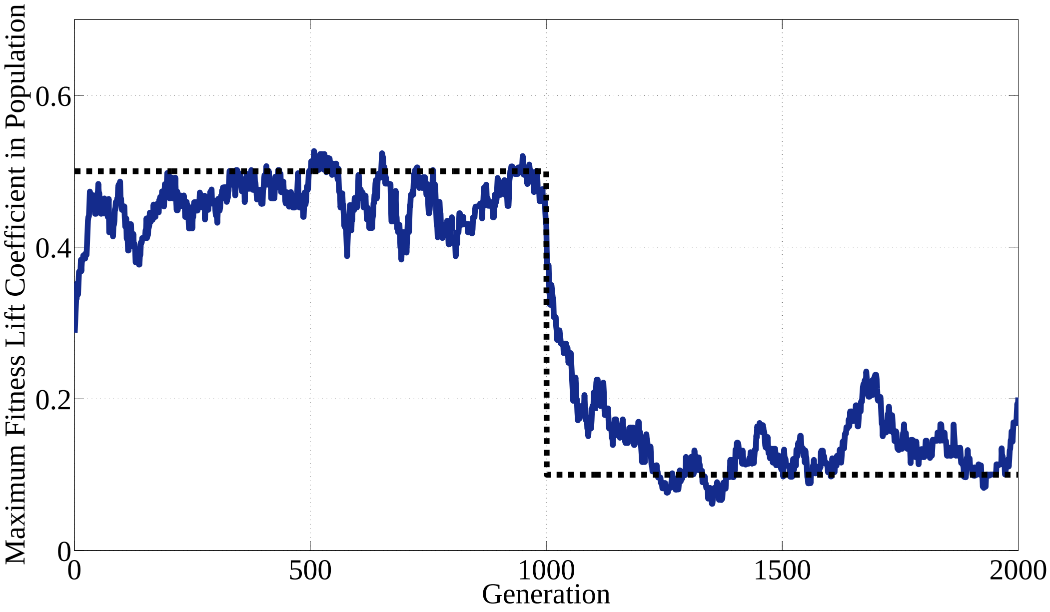


Figure 5.12: Target (dashed) and actual (solid) maximum fitness lift coefficient in the CGAFPS population per generation.

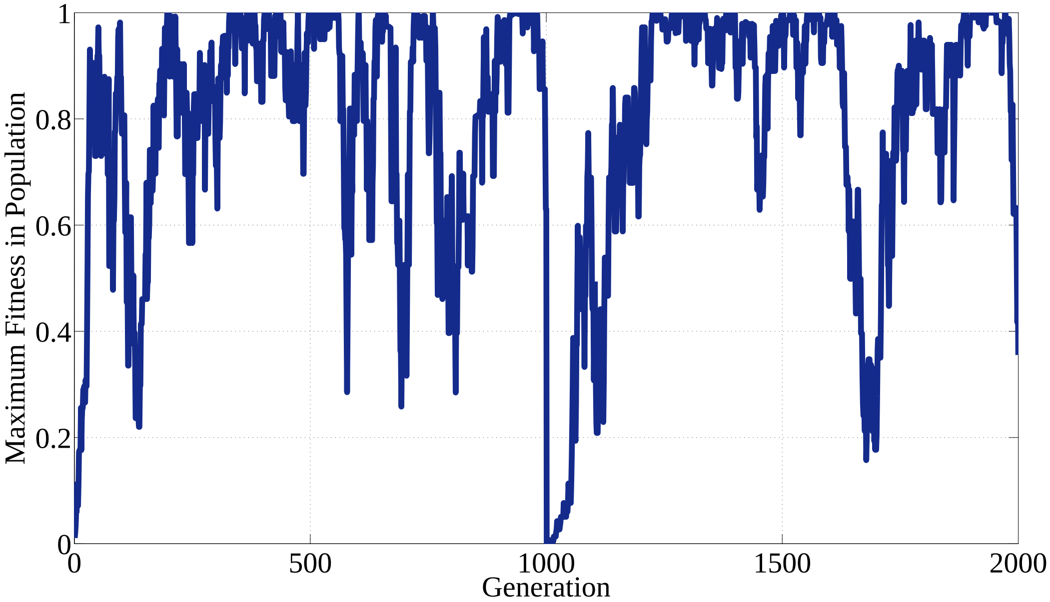


Figure 5.13: Maximum fitness value in the CGAFPS population per generation.

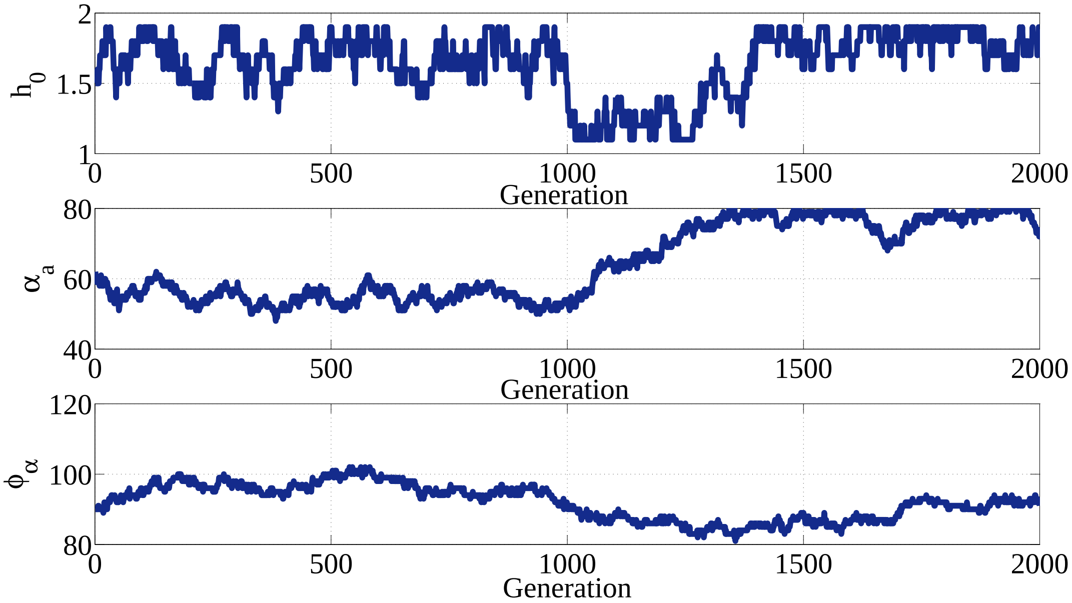


Figure 5.14: Flapping wing kinematic parameters corresponding to the maximum fitness lift coefficient in the CGAFPS population per generation.

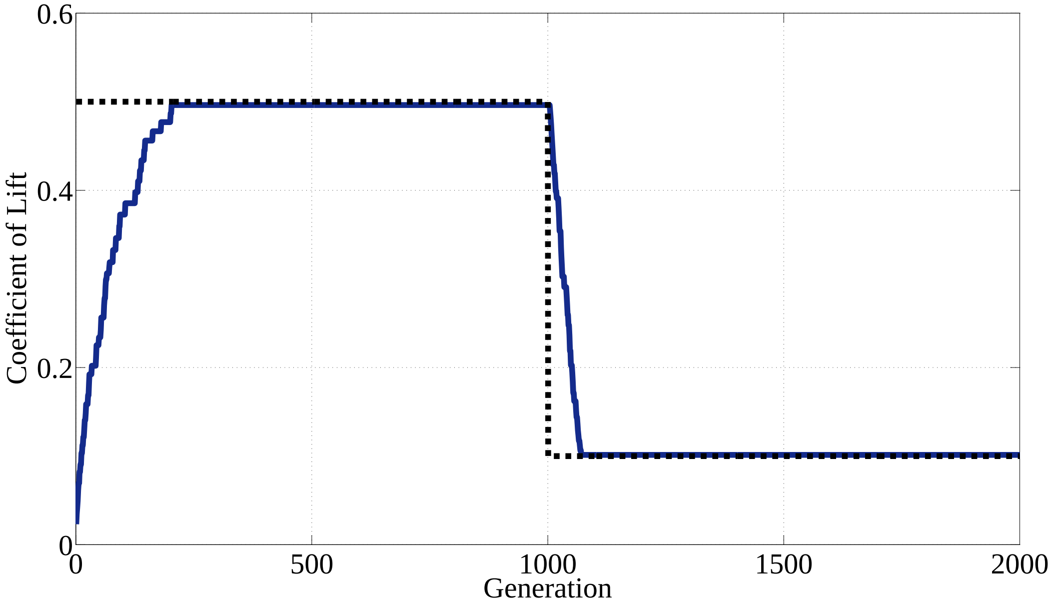


Figure 5.15: Target (dashed) and actual (solid) lift coefficients per generation with the (1+1)-ES.

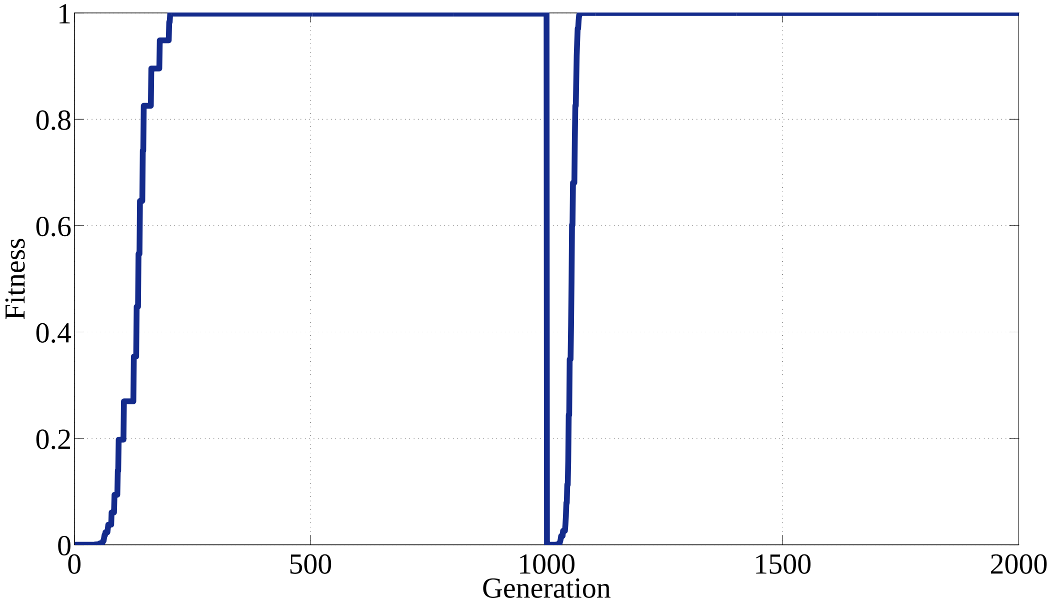


Figure 5.16: Fitness per generation with the (1+1)-ES.

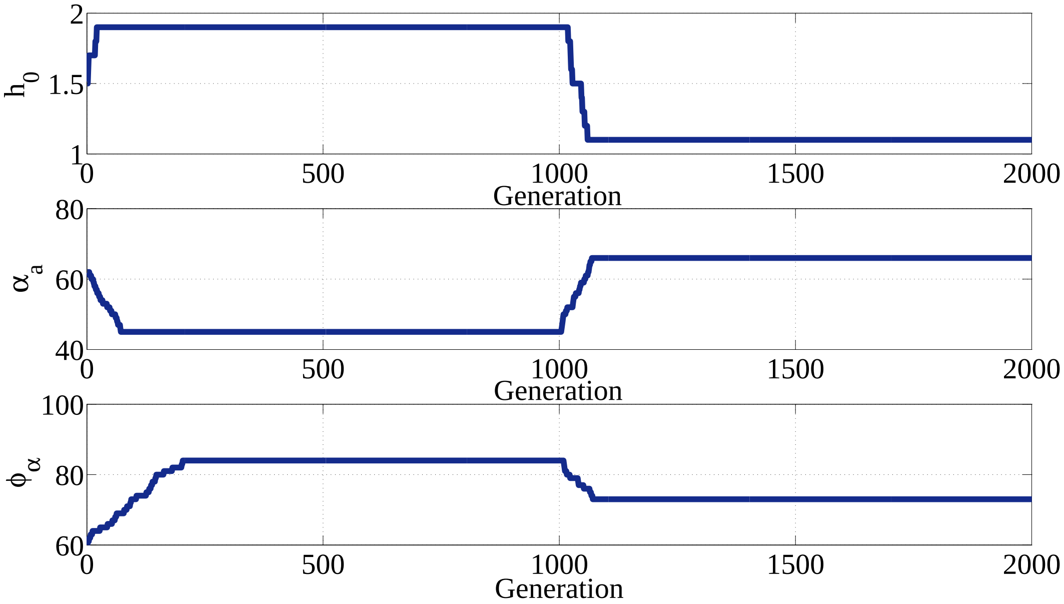


Figure 5.17: Flapping wing kinematic parameters with the (1+1)-ES.

Figures 5.18 to 5.20 visually compare the resilience characteristics of these two algorithms to the SEGS technique. The canonical genetic algorithm with fitness proportional selection exhibits resilient behavior, which is unsurprising since it is similar to a SEGS scheme with $N = 1$. The (1+1) evolutionary strategy behaves like a SEGS with a level of selectivity that exceeds 100.

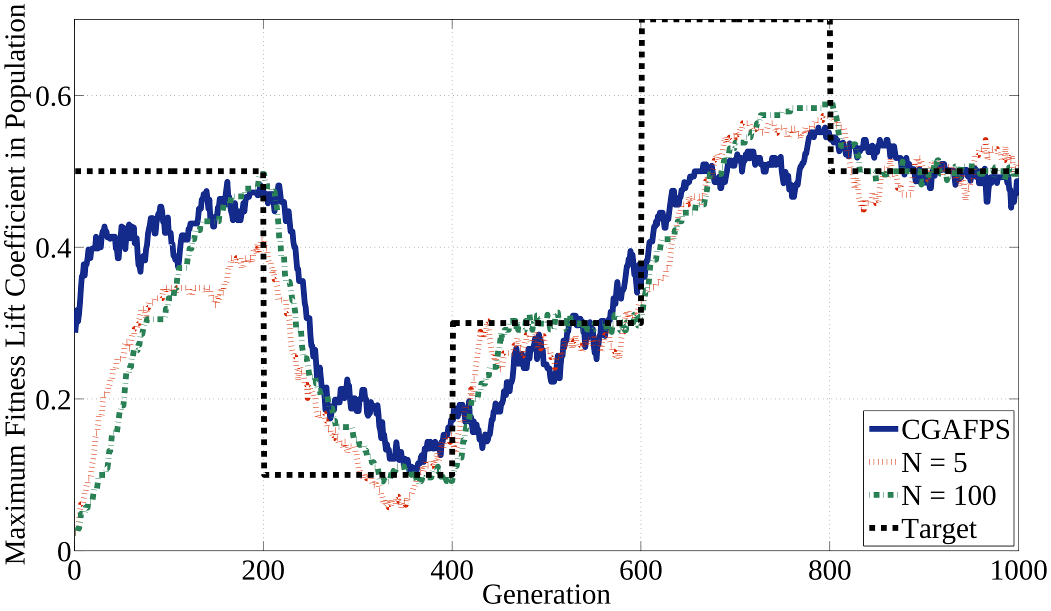


Figure 5.18: Comparison of the maximum fitness lift coefficient in the CGAFPS population and the SEGS lift coefficient per generation.

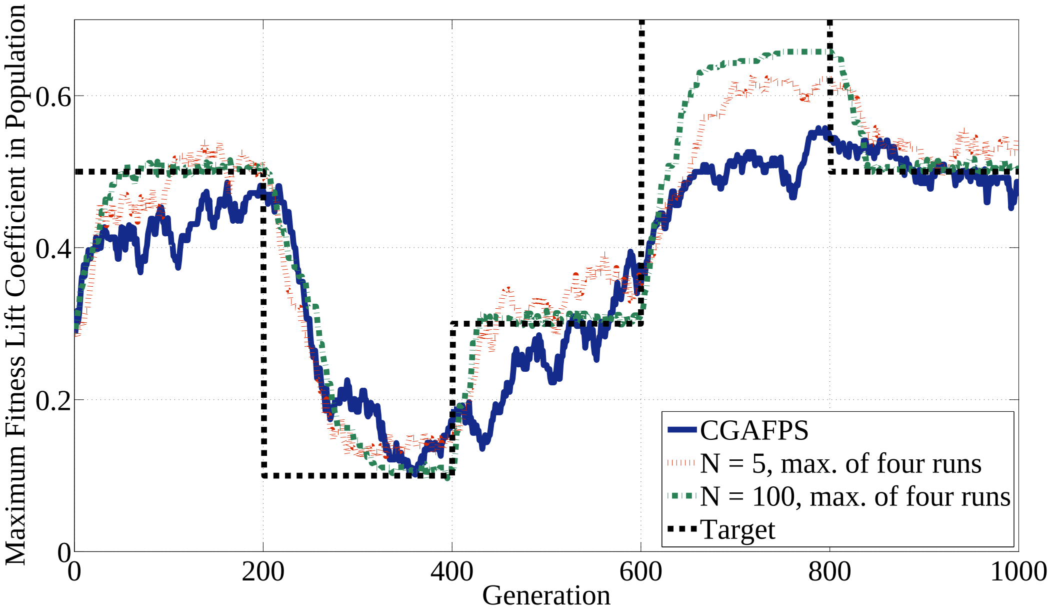


Figure 5.19: Comparison of the maximum fitness lift coefficient in the CGAFPS population and the maximum fitness lift coefficient of four SEGS runs per generation; the four SEGS runs were uniquely initialized with a member of the initial population of the CGAFPS.

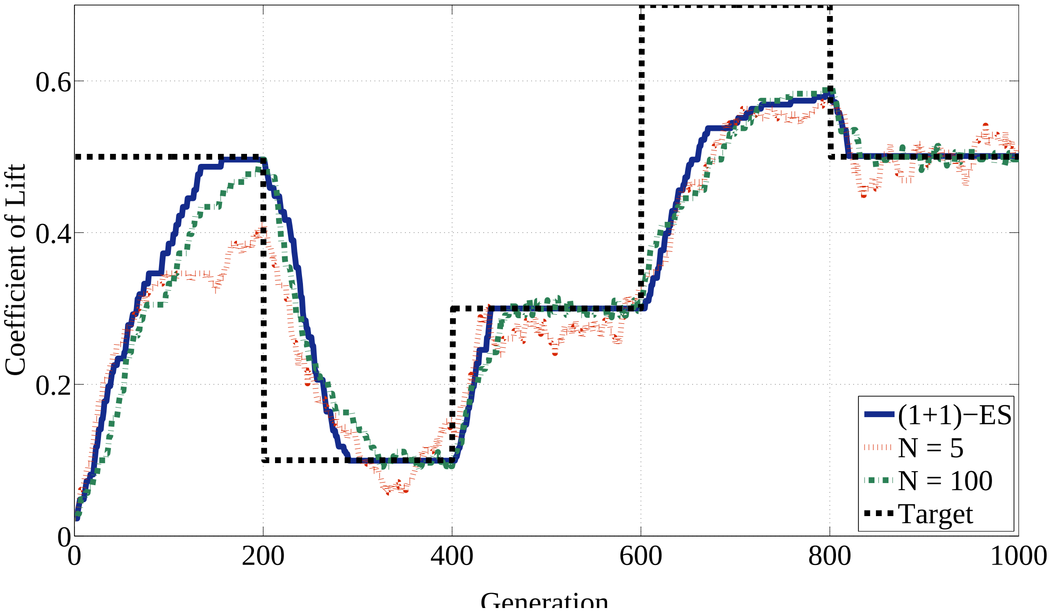


Figure 5.20: Comparison of the lift coefficients of the (1+1)-ES and the SEGS per generation.

5.5 Theodorsen-Garrick Model

This example utilizes the developed theory and a model embodied by unsteady flow equations to consider the standard reference-tracking problem in control systems within the context of flapping flight. The work here is different from [149–155] in that we evolve flapping wing parameters for trajectory tracking, instead of taking a control-theoretic approach. The chosen model outputs forces from which trajectories can be computed, and these trajectories are then analyzed by a SEGS. This approach, and the other differences from the previous example (forward motion of the flapping wing, incorporation of actual physics, and an addition to the literature), are a legitimate reason to include the application here.

The Theodorsen-Garrick model [156, 157] predicts the lift and thrust forces on a flat plate undergoing a prescribed flapping motion with various input kinematic parameters. This flapping motion is described by

$$h(t) = h_a(t) \sin\left(\omega\left(t\right) t + \phi_h\left(t\right)\right), \tag{5.11}$$

$$\alpha(t) = \alpha_a(t) \sin\left(\omega\left(t\right) t + \phi_\alpha\left(t\right)\right), \tag{5.12}$$

where $h_a(t) \in (0, 1]$ and $\alpha_a(t) \in [-0.5, 0.5]$ are the piecewise-constant amplitudes of flapping stroke height and angle of attack respectively, $\omega(t) \in (0, 1]$ is a piecewise-constant frequency, and $\phi_h(t) \in [-0.5, 0.5]$ and $\phi_\alpha(t) \in [-0.5, 0.5]$ are the piecewise-constant phase shift angles for flapping stroke height and angle of attack, respectively. The flapping motion described in (5.11)–(5.12) leads to the computation of lift and thrust forces through the equations stated in [157]. These forces determine the trajectory followed by the flapping wing; hence, the flapping flight motion problem: given a target trajectory (e.g., a constant altitude forward motion trajectory), find suitable flapping wing kinematic parameters that meet the target.

We utilize the following evolutionary generation system, (X, R, P, G), which is formulated as a random walk over a discretized search space.

- The set of cells, X, is the set of ordered pentuples $\left(h_a(t), \omega(t), \phi_h(t), \alpha_a(t), \phi_\alpha(t)\right)$, where

$$h_a(t) \in \{0.1, 0.2, 0.3, \ldots, 0.9, 1\}, \tag{5.13}$$

$$\omega(t) \in \{0.05, 0.1, 0.15, \ldots, 0.95, 1\}, \tag{5.14}$$

$$\phi_h(t) \in \{-0.5, -0.45, -0.4, \ldots, 0.45, 0.5\}, \tag{5.15}$$

$$\alpha_a(t) \in \{-0.5, -0.45, -0.4, \ldots, 0.45, 0.5\}, \tag{5.16}$$

$$\phi_\alpha(t) \in \{-0.5, -0.45, -0.4, \ldots, 0.45, 0.5\}. \tag{5.17}$$

- The set of resources, R, is the set $\{r_1, r_2, r_3, r_4, r_5, r_6, r_7, r_8, r_9, r_{10}\}$, with $r_i = \mathbf{e}_i$, $1 \le i \le 10$ (where $\mathbf{e}_i$ are the standard basis vectors for $\mathbb{R}^{10}$). This choice of resources facilitates the perturbation of one of the elements of a cell in either a positive or negative direction when an offspring is generated.

- The probability mass function on R, P, is the discrete uniform distribution. This choice of probability mass function ensures that the matrix of generation probabilities is symmetric.

- The generation function, G, applied to X as

$$G\left(\left(h_a(t), \omega(t), \phi_h(t), \alpha_a(t), \phi_\alpha(t)\right), r_i\right), 1 \le i \le 10, \tag{5.18}$$

is the pentuple given by

$$\begin{cases} 1\times 10^{-2} A r_i + \begin{bmatrix} h_a\left(t\right) \\ \omega\left(t\right) \\ \phi_h\left(t\right) \\ \alpha_a\left(t\right) \\ \phi_\alpha\left(t\right) \end{bmatrix}, \text{ where } A = \\ \begin{bmatrix} 10 & -10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 5 & -5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 5 & -5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 5 & -5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & -5 \end{bmatrix}, \\ \text{if } 0.1 < h_a\left(t\right) < 1,\ 0.05 < \omega\left(t\right) < 1, \\ \quad -0.5 < \phi_h\left(t\right) < 0.5,\ -0.5 < \alpha_a\left(t\right) < 0.5, \\ \quad -0.5 < \phi_\alpha\left(t\right) < 0.5, \\ \\ \left(h_a\left(t\right), \omega\left(t\right), \phi_h\left(t\right), \alpha_a\left(t\right), \phi_\alpha\left(t\right)\right), \text{ otherwise.} \end{cases} \tag{5.19}$$

The flapping wing parameters evolved by the SEGS are inputs for the Theodorsen-Garrick model, which outputs lift $L(\tau)$ and time-averaged-thrust $T(\tau)$ over time τ. These forces are in turn inputs for the following double-integrator, unit-mass wing trajectory dynamics,

$$\begin{bmatrix} \dot{x}(\tau) \\ \dot{y}(\tau) \\ \dot{v}_x(\tau) \\ \dot{v}_y(\tau) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x(\tau) \\ y(\tau) \\ v_x(\tau) \\ v_y(\tau) \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} L(\tau) \\ T(\tau) \end{bmatrix}, \tag{5.20}$$

where $(x(\tau), y(\tau))$ is the trajectory of the center of mass of the flapping wing. This trajectory is sampled ν times, yielding $(x(k), y(k))$, $1 \leq k \leq \nu$. For each $x(k)$, the target $y_{des}(k)$ is computed. Let

$$AvgDistance(t) = \frac{\sum\limits_{k=1}^{\nu} |y_{des}(k) - y(k)|}{\nu} \tag{5.21}$$

be the mean difference between the target and current trajectories. Since the objective is to track the target, we use the following fitness function for the SEGS,

$$F(h_a(t), \omega(t), \phi_h(t), \alpha_a(t), \phi_\alpha(t)) = \exp\left(-(0.1AvgDistance(t))^2\right). \tag{5.22}$$

5.6 Theodorsen-Garrick Model Results

A sample initial trajectory together with a trajectory obtained from that sample after 200 generations with $N = 5$ are plotted in Figure 5.21, where the trajectories are depicted over the same period of time. The figure shows that the evolved kinematic parameters reduce altitude excursions away from the target trajectory by a factor of four while utilizing roughly the same amount of time-averaged-thrust that was specified by the initial set of kinematic parameters. Moreover, the average evolved trajectory tracks the constant altitude desired trajectory, while the average initial trajectory does not.

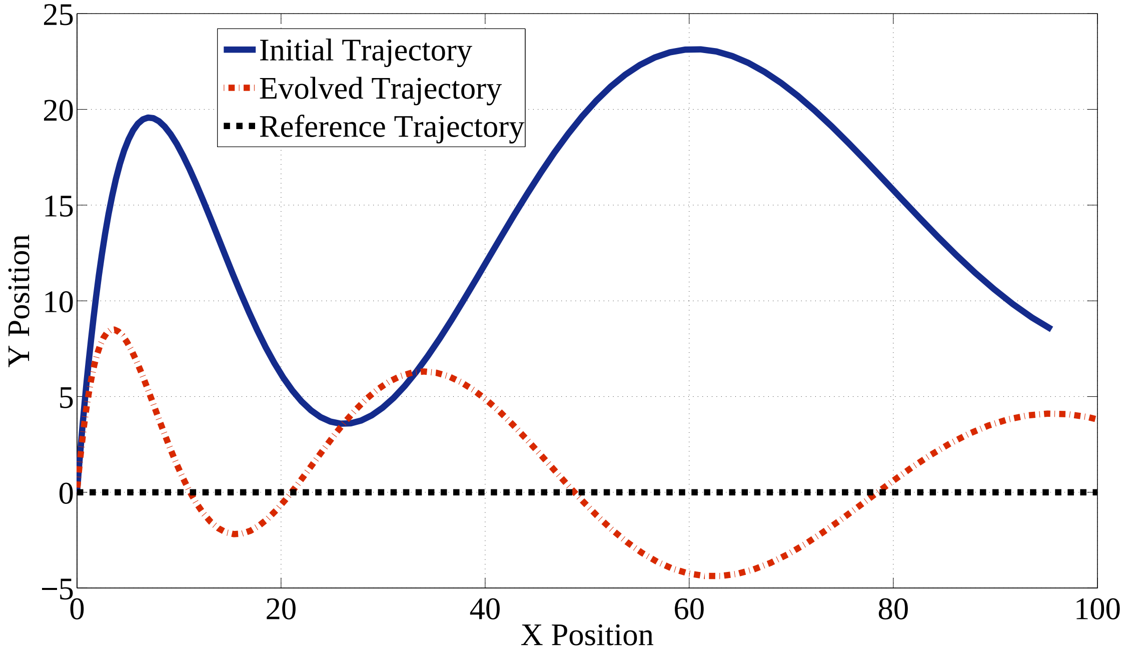


Figure 5.21: Target trajectory (dashed), initial trajectory (solid) and the 200th evolved (dashed-dotted) trajectory.

The scheme requires, on average, 2 minutes 34 seconds to compute the output of 200 generations while running in MATLAB on a 2.50 GHz dual-core processor laptop with 4.00 GB of RAM and the Windows Vista operating system.

CHAPTER 6

EVOLUTION OF ROBUST CONTROL

6.1 Problem Description

This chapter applies the theory developed in Chapter 4 to controller design for a dynamic system model of a xerographic process. This model was developed with collaborators in industry, and refers to a state-of-the-art technological system. Utilizing the scheme developed in this dissertation, the problem tackled in this chapter is the optimization of control gains such that acceptable disturbance-rejection performance is achieved by the system despite internal parameter fluctuations. For proprietary reasons, all data in this chapter have been represented with symbols.

Xerographic images exhibit many kinds of defects, including variations in the print of a solid color. These variations are due, in part, to the effect of disturbances entering the system at different stages of the print process. Color variations may also result from fluctuations in the settings of a stage. A parameter, *DeltaE*, exists to quantify the output color variation relative to a print that is made in the absence of disturbances and at nominal settings. The human eye is incapable of perceiving color variations when the absolute value of *DeltaE* is less than or equal to one. Hence, the controller is required to achieve print quality such that *DeltaE* is nominally zero, but no more than ±1.

6.2 Xerographic System Model

Consider the system block diagram in Figure 6.1, where the plants, P_1 and P_2, and the plant output, v_3, are subject to external disturbances $d_1 \in [d_{1i}, d_{1f}]$ and $d_2 \in [d_{2i}, d_{2f}]$ as shown. The input r is a reference signal, v_1, v_2 and v_3 are intermediate signals, and y is the output signal $DeltaE$. Control signals u_1 and u_2 utilize the measured signals v_2 and y, and control gains K_1 and K_2.

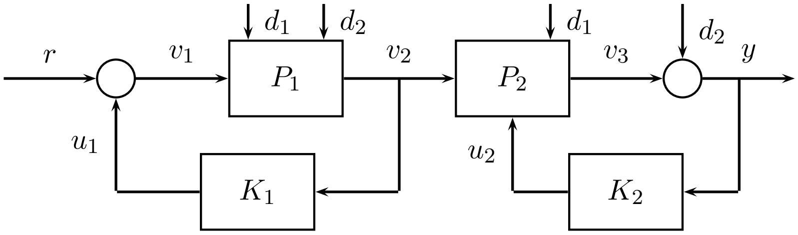


Figure 6.1: Block diagram of the example xerographic dynamic system.

We utilize the following evolutionary generation system, (X, R, P, G), which is formulated as a random walk over a discretized search space.

- The set of cells, X, is the set of the set of ordered pairs (K_1, K_2) where

$$K_1 \in \{-20, -19.75, -19.5, \ldots, 19.75, 20\}, \tag{6.1}$$

$$K_2 \in \{-20, 19.75, -19.5, \ldots, 19.75, 20\}. \tag{6.2}$$

- The set of resources, R, is the set $\{r_1, r_2, r_3, r_4\}$, with $r_i = \mathbf{e}_i,\ 1 \leq i \leq 4$ (where $\mathbf{e}_i$ are the standard basis vectors for $\mathbb{R}^4$). This choice of resources facilitates the perturbation of one of the elements of a cell in either a positive or negative direction when an offspring is generated.

- The probability mass function on R, P, is the discrete uniform distribution. This choice of probability mass function ensures that the matrix of generation probabilities is symmetric.

- The generation function, G, when applied to $(K_1, K_2) \in X$ using resource $r_i \in R$, $1 \leq i \leq 4$, yields

$$\begin{cases} \begin{bmatrix} 0.25 & -0.25 & 0 & 0 \\ 0 & 0 & 0.25 & -0.25 \end{bmatrix} r_i + \begin{bmatrix} K_1 \\ K_2 \end{bmatrix}, \\ \text{if } -20 < K_1 < 20, \; -20 < K_2 < 20, \\ \\ (K_1, K_2), \; \text{otherwise.} \end{cases} \tag{6.3}$$

The desired output is $y_{des} = DeltaE_{des} = 0$, with an acceptable tolerance of ±1. Therefore, a suitable fitness function is

$$F = \exp\left(-\left(K_f\left(y_{des} - y\right)\right)^2\right), \tag{6.4}$$

where

$$K_f = 1. \tag{6.5}$$

6.3 Results

A sample run of the SEGS when $N = 5$ is depicted in Figures 6.2 to 6.5 for fixed disturbances d_1 and d_2. A pair of control gains that achieves satisfactory performance is found within 50 generations. To demonstrate disturbance rejection, the disturbances are varied after 50 generations and the scheme is quickly able to find a new pair of gains that achieves an acceptable output.

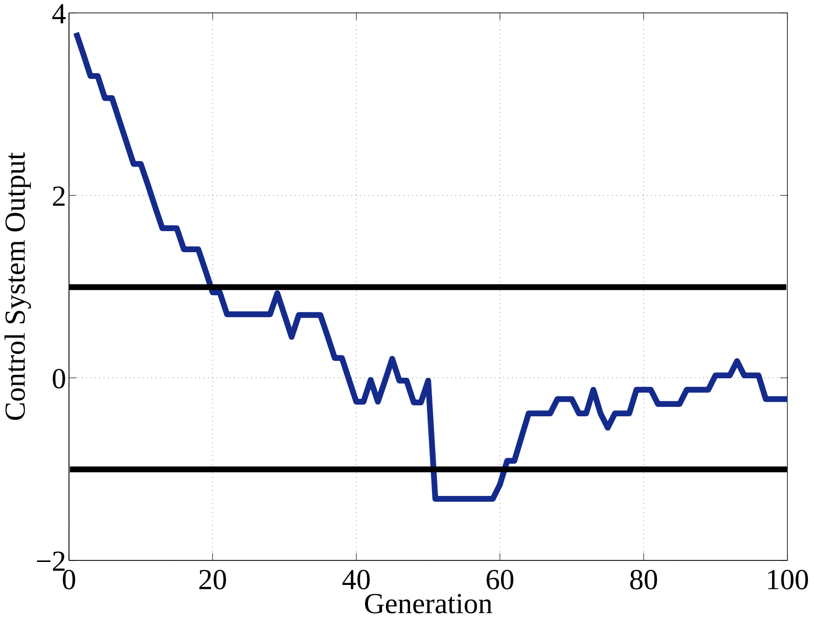


Figure 6.2: Satisfactory output is maintained despite disturbance changes at generation 50.

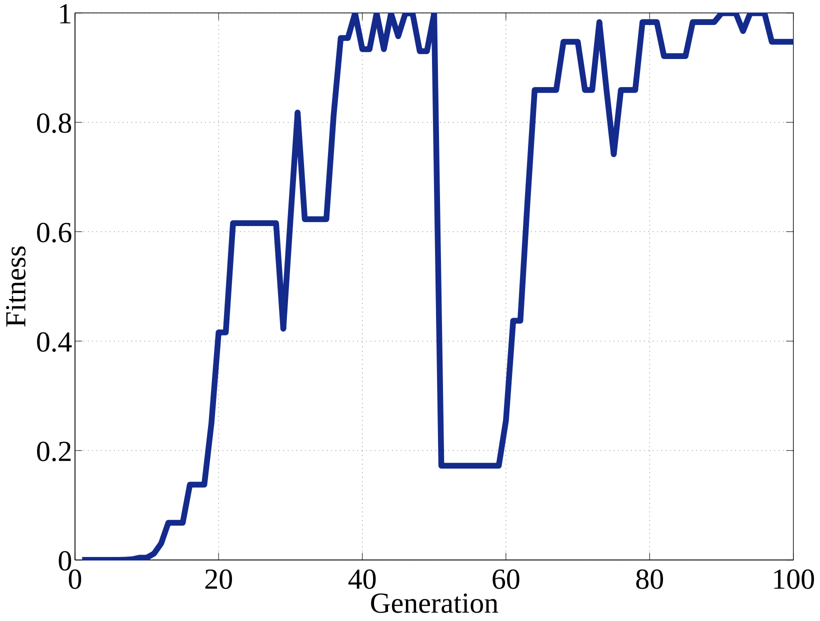


Figure 6.3: Fitness of the control gains per generation.

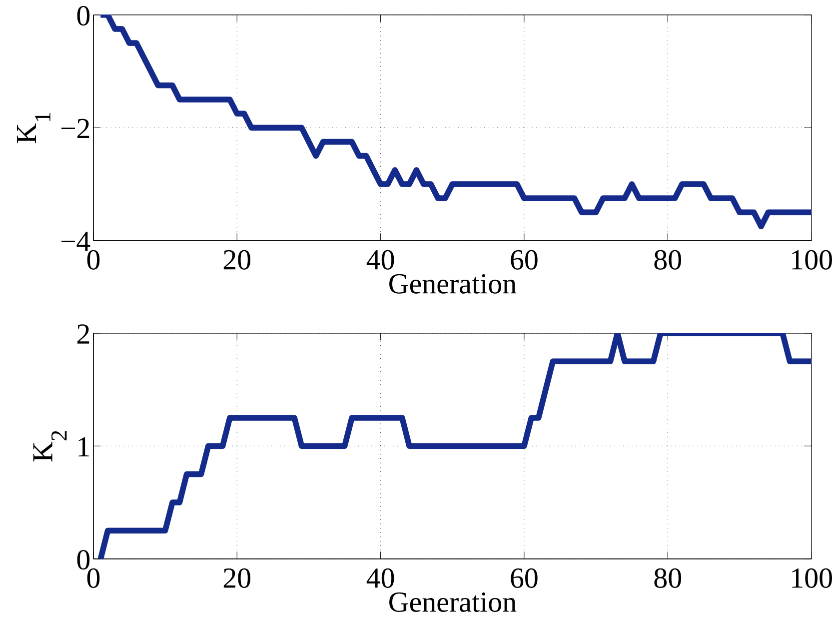


Figure 6.4: Control gain pairs per generation.

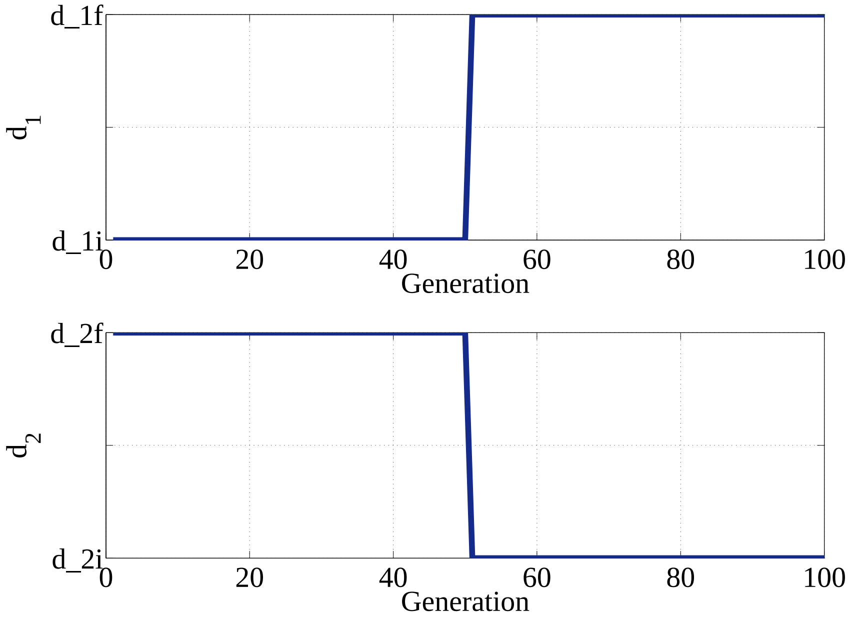


Figure 6.5: Disturbance variations at generation 50.

Similarly, the SEGS is resilient to internal model variations, and this is depicted in Figures 6.6 to 6.8 for fixed disturbances d_1 and d_2, $N = 5$, and an internal parameter change at the 50th generation.

Typically, the scheme averages 21 seconds to compute the output of 100 generations while running in MATLAB on a 1.4 GHz single processor desktop computer with 1 GB of RAM.

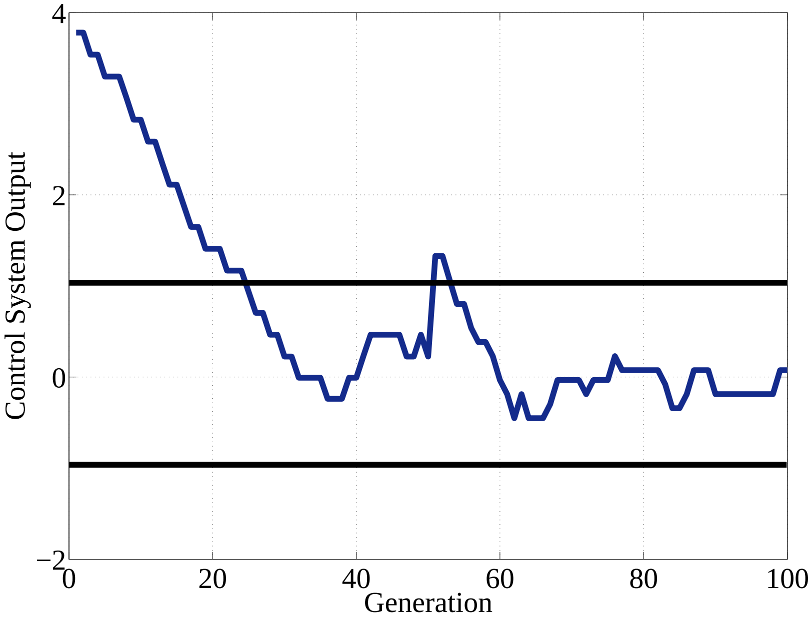


Figure 6.6: Satisfactory output is maintained despite an internal parameter variation at generation 50.

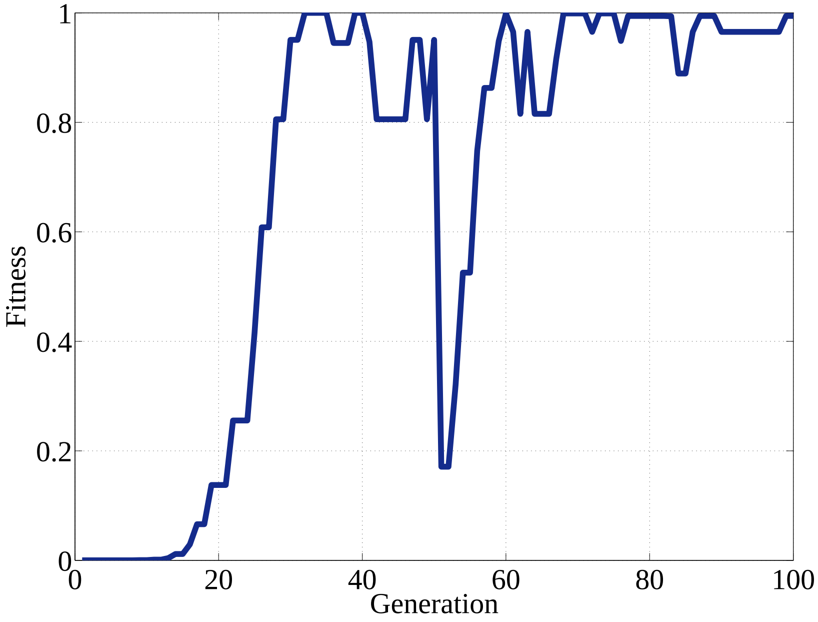


Figure 6.7: Fitness of the control gains per generation.

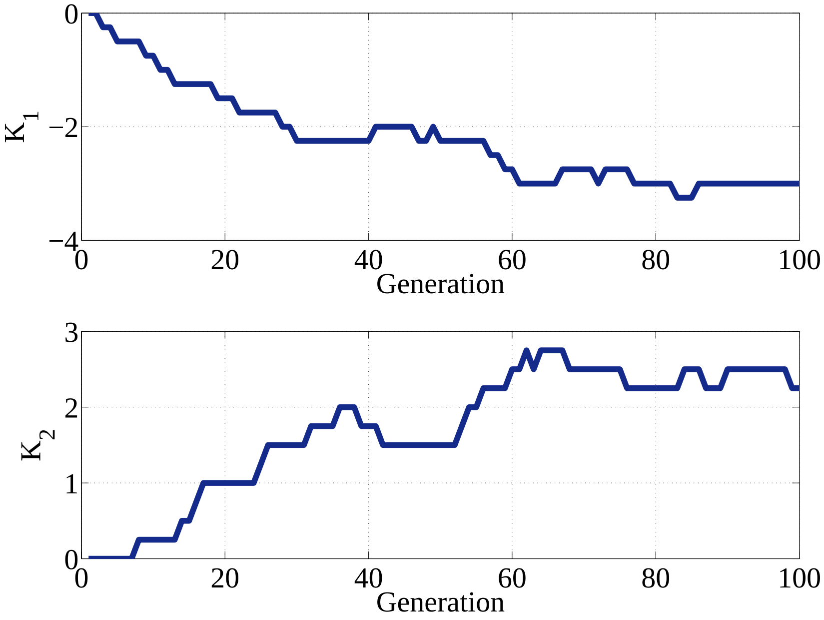


Figure 6.8: Control gain pairs per generation.

CHAPTER 7

EVOLUTION OF RESILIENT GRAMMAR INFERENCE

7.1 Problem Description

The problem of *resilient grammar inference* seeks to design the behavior of a deterministic finite-state automaton [158] so that:

1. with high fidelity, it accepts those and only those sequences of symbols satisfying given rules, based on a training set of sequences of symbols whose acceptability is known, and
2. when the rules constraining the sequences of symbols change (and the training set changes accordingly), the automaton resiliently adapts to the new rules.

The problem of resilient grammar inference is a generalization of the standard problem of grammar inference [110, 159] (also called grammar induction or grammar recognition), which addresses only requirement 1) above. Grammar inference has long been acknowledged as a hard problem [160–162] with relevance to theoretical computer science [163], computational learning theory [164], machine learning [108, 109], pattern recognition [165, 166] and artificial intelligence [2] (which is itself a key enabling technology of robotics [167]), and has been treated in many papers (see, for instance, the survey papers [168–173]). The traditional emphasis [173] has been on learning regular grammars or deterministic finite-state automata, with methods for addressing the problem including data-dependent approaches [160, 174], lattice representations [175, 176], evidence-driven and data-driven heuristics [177, 178], artificial intelligence techniques [179] and genetic algorithms [180]. In practice, however, a grammar-inferring finite-state automaton may operate in a time-varying or dynamic environment to which it must adapt; this adaptability to an uncertain and changing environment is a requirement of intelligence [25]. Surprisingly, the problem of resilient grammar inference has received relatively little attention in the literature, with the exception of [28]. Of course, the sequential repetition of any method for designing a grammar-inferring finite-state automaton yields a resilient design method. However, such sequential repetitions are computationally expensive. Hence, the goal of this chapter is to present a resilient and inexpensive method for designing the behavior of a grammar-inferring finite-state automaton.

7.2 Finite-State Automata Model

We are interested in finding deterministic finite-state automata [158] capable of correctly accepting or rejecting training data strings, thereby inferring a regular grammar. Although any regular grammar may be chosen for testing the inference capability of a SEGS, for simplicity, the grammar that we infer is the set of all strings in $\{1, 0\}^*$ that have an even number of zeros. This simplicity allows us to identify species of grammar-inferring automata, physiological features common to individuals within a species, and morphological features common to all such species.

We take a deterministic finite-state automaton to be a quintuple $(Q, \Sigma, s, A, \delta)$, where Q is a finite set of states, $\Sigma = \{1, 0\}$ is the binary alphabet, $s \in Q$ is the start state, $A \subseteq Q$ is the set of accept states, and $\delta : Q \times \Sigma \rightarrow Q$ is the transition function. Since the set of deterministic finite-state automata is enumerable [158], each automaton may be mapped to a unique natural number. A ν-state automaton has

ν possibilities for the start state, 2^ν choices of the accept state(s), and $\nu^{2\nu}$ feasible transitions from state to state. Thus, there are a total of $2^\nu \nu^{2\nu+1}$ arrangements of a ν-state automaton. For instance, there are two arrangements of a 1-state automaton (where $|A| = 0$ and $|A| = 1$), which may be mapped to the natural numbers 1 and 2, respectively. Similarly, there are 128 arrangements of a 2-state automaton, which may be mapped to the numbers 3 through 130. This process can be continued *ad infinitum.*

The number 24 corresponds to the deterministic finite-state automaton in Figure 7.1, a minimal finite-state automaton that infers the chosen grammar. To interpret this number as a finite-state automaton, consider that 24 is in the interval between 3 and 130, which implies a 2-state automaton. Since 24 is in the first half of the interval, the first state is the start state. Next, because 24 belongs to the second sub-interval of sixteen numbers ($16 = 2^{2\cdot 2}$), $|A| = 1$, and the first state is an accept state. The remainder (of six), when expressed in a base equal to the number of states, indicates the state-to-state transition for each letter in the alphabet.

$$Q = \{q_0, q_1\},\ \Sigma = \{1, 0\},\ s = \{q_0\},\ A = \{q_0\}$$

q	σ	$\delta(q,\sigma)$
q_0	1	q_0
q_0	0	q_1
q_1	1	q_1
q_1	0	q_0

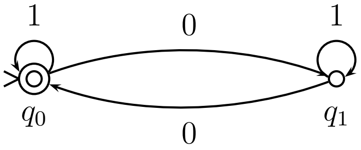


Figure 7.1: Minimal deterministic finite-state automaton that accepts a string with an even number of zeros.

Therefore, in this application the set of natural numbers is the set of genotypes, the set of deterministic finite-state automata is the set of phenotypes, and the genotype-phenotype mapping is described by the mapping above. The pseudocode to evaluate this mapping is provided in Appendix B. This mapping is bijective, and therefore differs from the non-injective genotype-phenotype mapping in nature [105].

We utilize the following evolutionary generation system, (X, R, P, G).

- The set of cells, X, is the set of natural numbers $\{1, 2, \ldots, 1 \times 10^{100}\}$, where a large, finite upper bound is required in accordance with Section 3.1, although it is not necessary in practice.

- The set of resources, R, is the set $\{r_1, r_2, \ldots, r_{41}, r_{42}\}$, with

$$r_i = \begin{cases} 2^{i-1}, & 1 \leq i \leq 21, \\ -\left(2^{i-22}\right), & 22 \leq i \leq 42. \end{cases} \tag{7.1}$$

- The probability mass function on R, P, is the discrete uniform distribution. This choice of probability mass function ensures that the matrix of generation probabilities is symmetric.

- The generation function, G, when applied to $x \in X$ using resource $r \in R$, yields

$$G\left(x, r\right) = \begin{cases} x + r, & \text{if } x + r \geq 1, \\ x, & \text{otherwise.} \end{cases} \tag{7.2}$$

Note that the evolutionary generation system above describes a base-2 exponential random walk on the set of natural numbers, X. An alternative formulation is to represent the progenitor genotype natural number as a binary string, and perturb one bit in this string to produce an offspring.

We design the SEGS so that one-step transitions of at least 1,000,000 between natural number genotypes is possible. This requires the large, 42 element resource set above. It is certainly feasible to design the evolutionary generation system so that fewer resources are utilized; for example, a base-10 exponential random walk on the set of natural numbers, X, requires only thirteen resources. Unfortunately, fewer cells are reachable through G and R in one step when the evolutionary generation system is a base-10 exponential random walk.

As a result of the genotype-phenotype mapping defined above, the problem of grammar inference is a particular case of an unbounded search in an ordered list [181] for a satisfactory natural number. Unlike the standard problem treated in the literature, multiple natural numbers are satisfactory here. This is because there are multiple finite-state automata that can infer a given regular grammar; however, these phenotypes can be reduced to a minimal finite-state automaton through a minimization algorithm [163]. It is well known [181] that a base-2 exponential unbounded search is an efficient approach to finding the desired natural number, although there are other, more involved, algorithms that are nearly optimal [181–183].

The fitness of a phenotype is evaluated by its successful classification of a set of training data. This training data consists of 1,024 (a design parameter) randomly selected binary strings of length between 1 and 16 bits. The SEGS has access to the correct classification of the binary strings, and this correct classification may be time-varying. Due to the randomness of the training data selection, both acceptable and unacceptable strings are included in the training set, satisfying the requirement that positive and negative data samples be used for grammar inference [159].

Let $FractionCorrect(x)$ denote the fraction of training data strings that is correctly accepted or rejected by the finite-state automaton x. We use the fitness function

$$F\left(x\right) = \exp\left(-\left(K_f\left(1 - FractionCorrect(x)\right)\right)^2\right), \tag{7.3}$$

where

$$K_f = 2. \tag{7.4}$$

The finite-state automata that correctly classify the training data have

$$FractionCorrect(x) = 1, \tag{7.5}$$

and therefore have maximal fitness. For a SEGS with bounded fitness, we refer to a cell as *viable* if it is maximally fit; otherwise, it is *non-viable*. The viable finite-state automata form a subset of X, and coupling this subset with the generation transitions that produce a viable descendant from a viable ancestor through other viable cells gives us a subgraph of the directed graph representation of the SEGS. This subgraph may have reducible classes of viable, weakly-connected cells; we call each such class a *species*, i.e., a species is a set of viable cells that is arc-wise connected through G and R.

Note that determining whether two viable cells belong to the same species can be accomplished by using the Breadth First Search (BFS) algorithm [15] for example, which has run-time complexity that is linear with respect to the sum of the number of cells and number of generation transitions between cells. Also, note that reachability and irreducibility of a SEGS do not guarantee uniqueness of species: the set of viable cells may be non-reachable and reducible, and therefore consist of disconnected components, which are the distinct species.

We define *speciation* as the phenomenon whereby evolvants, generated by the dynamics (3.3), successively reside in a species for a number of generations, leave the species by becoming non-viable, then revisit a (possibly different) species, and so on. We emphasize that speciation is a random phenomenon — each simulation of the above SEGS realizes this phenomenon to a different degree.

Speciation in biology is defined as the evolution of two or more distinct species from a single ancestral species [105]. Our notion captures the biological definition of speciation in the following sense: when the set of viable cells is disconnected, the ergodicity property (see Chapter 4) guarantees that regardless of the ancestor, the descendant will, in time, visit all of the species. Therefore, any ancestor generates progeny that contains individuals from different species.

7.3 Results

7.3.1 Grammar Inference

The results of a typical sample run of the selective evolution scheme when $N = 5$ are presented in Figures 7.2 and 7.3. The figures depict that a finite-state automaton that infers the chosen grammar is evolved within 70 generations. This finite-state automaton correctly accepts or rejects all 1,024 training data strings and therefore has a maximal fitness value of one. At generation 70 for instance, the evolved finite-state automaton has genotype 1,715,127 and phenotype illustrated in Figure 7.4. It is easy to see how this finite-state automaton reduces to that of Figure 7.1: in Figure 7.4, there does not exist a transition to state q_0, which implies that this state can be deleted. It follows that states q_2 and q_3 may then be combined into a single state. The resultant deterministic finite-state automaton is exactly the minimal one with genotype 24.

Typically, the scheme averages 28 seconds to compute the output of 500 generations while running in MATLAB on a 2.50 GHz dual-core processor laptop with 4.00 GB of RAM and the Windows Vista operating system.

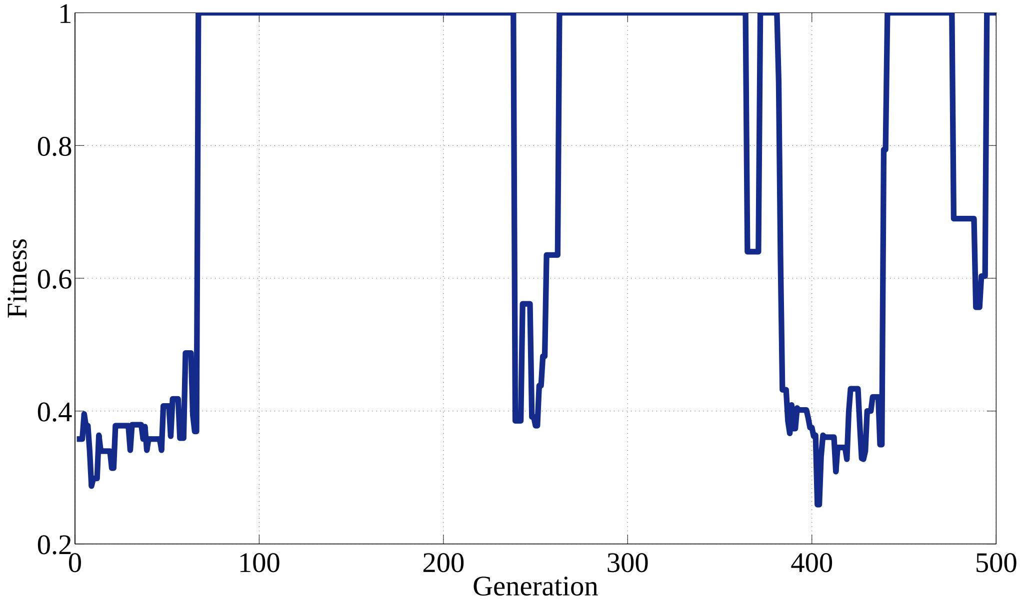


Figure 7.2: Fitness of the evolved deterministic finite-state automaton per generation.

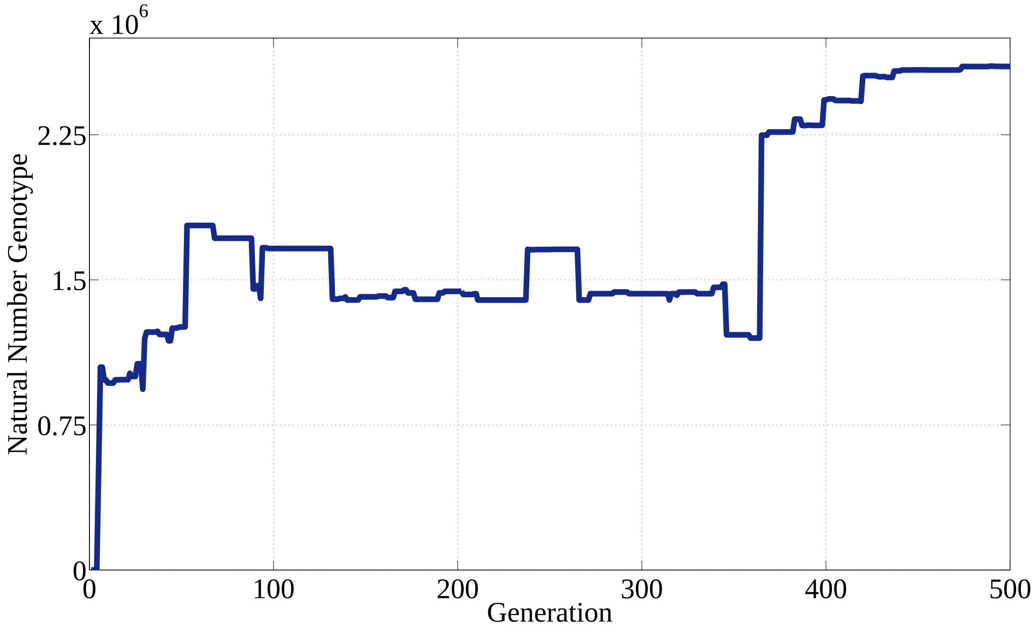


Figure 7.3: Natural number genotypes per generation.

$$Q = \{q_0, q_1, q_2, q_3\},\ \Sigma = \{1, 0\},\ s = \{q_1\},\ A = \{q_0, q_1\}$$

q	σ	$\delta(q,\sigma)$
q_0	1	q_3
q_0	0	q_2
q_1	1	q_1
q_1	0	q_2
q_2	1	q_3
q_2	0	q_1
q_3	1	q_3
q_3	0	q_1

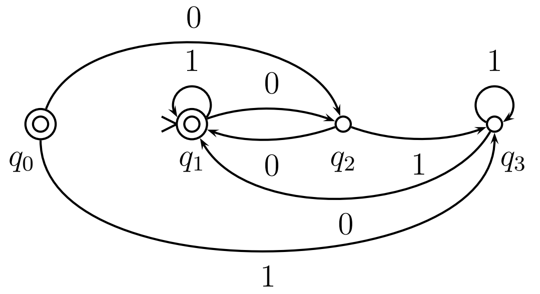


Figure 7.4: An evolved deterministic finite-state automaton that accepts a string with an even number of zeros.

7.3.2 Speciation

Inspection of Figures 7.2 and 7.3 reveals that there is more than one viable genotype (or deterministic finite-state automaton). One set of viable genotypes is found between generations 67 and 238, another set from generations 263 through 364, and yet more sets between generations 372 and 381, between generations 441 and 476, and between generations 495 and 500. The natural number genotypes in these sets are listed in Table 7.1. The figures and table demonstrate that speciation takes place as a consequence of selective evolutionary generation.

Table 7.1: Speciation Caused by Selective Evolutionary Generation

Set Number (Species Number)	Generation Range	Set Members	Set Cardinality	Common Members
1 (1)	67–238	1395571, 1395573, 1395575, 1395635, 1395719, 1395847, 1399731, 1399735, 1399815, 1403895, 1403911, 1407991, 1408023, 1412087, 1412103, 1412215, 1412231, 1416183, 1424307, 1428403, 1432499, 1432503, 1432631, 1440691, 1440759, 1440791, 1440823, 1449015, 1453047, 1469431, 1657717, 1661959, 1666039, 1666055, 1715127, 1715191, 1715383, 1780919	38	1395847, with Set 2
2 (1)	263–364	1199175, 1199239, 1215623, 1395639, 1395847, 1420423, 1428407, 1428535, 1428551, 1428567, 1428615, 1428631, 1428679, 1436807, 1436823, 1436839, 1436871, 1461383, 1477767, 1657783	20	1395847, with Set 1
3 (1)	372–381	2264151	1	-
4 (1)	441–476	2583970, 2584482, 2584674, 2584738, 2584742, 2584930, 2586790, 2602918, 2603174	9	-
5(1)	495–500	2602914	1	-

7.3.3 Physiology

The physical structures of an evolvant, its *physiology* [105], vary between species but are common within any given species. Here, we identify the physiological characteristics of a finite-state automata species that infers the chosen grammar. A BFS search reveals that all of the genotypes listed in Table 7.1 belong to the same species; hence, we consider the phenotypes of the elements of all five sets. We also note that the evolutionary process in nature does not affect the evolved physiology. We illustrate this aspect by including the grammar-inferring finite-state automaton with genotype 5,105 (Figure 7.5), also a member of the same species, in our discussion. This finite-state automaton emerged in a different selective evolutionary generation trial, and is not a member of any of the sets in Table 7.1.

$$Q = \{q_0, q_1, q_2\},\ \Sigma = \{1, 0\},\ s = \{q_0\},\ A = \{q_0, q_2\}$$

q	σ	$\delta(q, \sigma)$
q_0	1	q_2
q_0	0	q_1
q_1	1	q_1
q_1	0	q_0
q_2	1	q_2
q_2	0	q_1

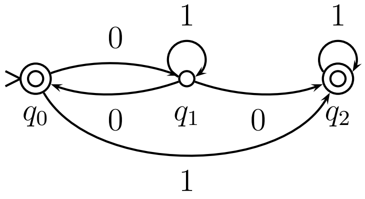


Figure 7.5: Deterministic grammar-inferring finite-state automaton evolved in a separate trial with genotype 5,105.

At first glance, the state transition diagrams of all three finite-state automata in Figures 7.1, 7.4, and 7.5 are quite different. However, the following characteristics are common, and indeed fundamental, if a finite-state automaton is to recognize the chosen grammar.

1. A zero bit input causes a state transition from an accept state to a reject state, and vice-versa.

2. A one bit input causes a state transition from an accept state to another accept state, and from a reject state to another reject state.

3. An exception to the above occurs when the start state is a reject state, and this exception affects only the start state transitions. When the start state is a reject state, a first bit input of zero causes a state transition to another reject state. If the first bit input is a one, a state transition is made from the start state to an accept state. In both cases, the start state is never transitioned to again.

These characteristics are easy to predict and identify due to the simplicity of the chosen grammar. Yet the point remains that selective evolutionary generation mimics biological evolution, yielding cells that have common physiology.

7.3.4 Resilience

Suppose that at generation 151 in Figures 7.2 and 7.3, the grammar to be inferred changes to the set of all strings in $\{1, 0\}^*$ that have an even number of ones. A minimal deterministic finite-state automaton that recognizes the new grammar now corresponds to the number 27, and is drawn in Figure 7.6.

Figures 7.7 and 7.8 illustrate a sample run where the acceptability of the training data set changes at generation 151. As the figures demonstrate, the SEGS is resilient and recovers to find a viable finite-state automaton within 7 generations. In fact, a new species is evolved; the viable genotypes found in this simulation are listed in Table 7.2.

$$Q = \{q_0, q_1\},\ \Sigma = \{1, 0\},\ s = \{q_0\},\ A = \{q_0\}$$

q	σ	$\delta(q,\sigma)$
q_0	1	q_1
q_0	0	q_0
q_1	1	q_0
q_1	0	q_1

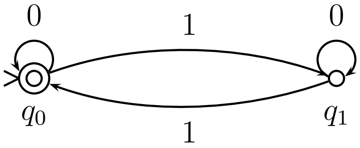


Figure 7.6: Minimal deterministic finite-state automaton that accepts a string with an even number of ones.

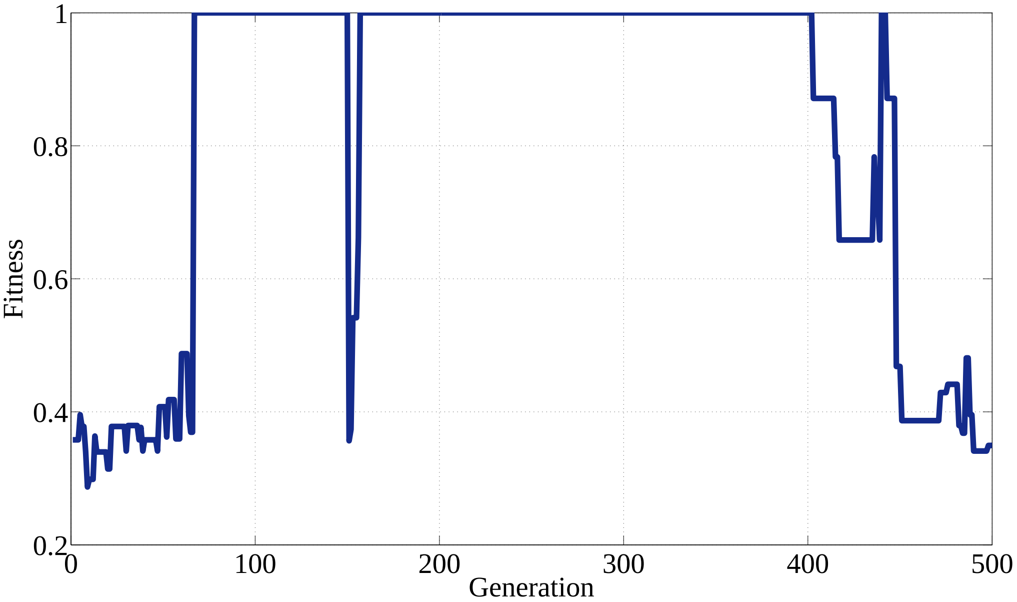


Figure 7.7: Fitness of the evolved deterministic finite-state automaton per generation.

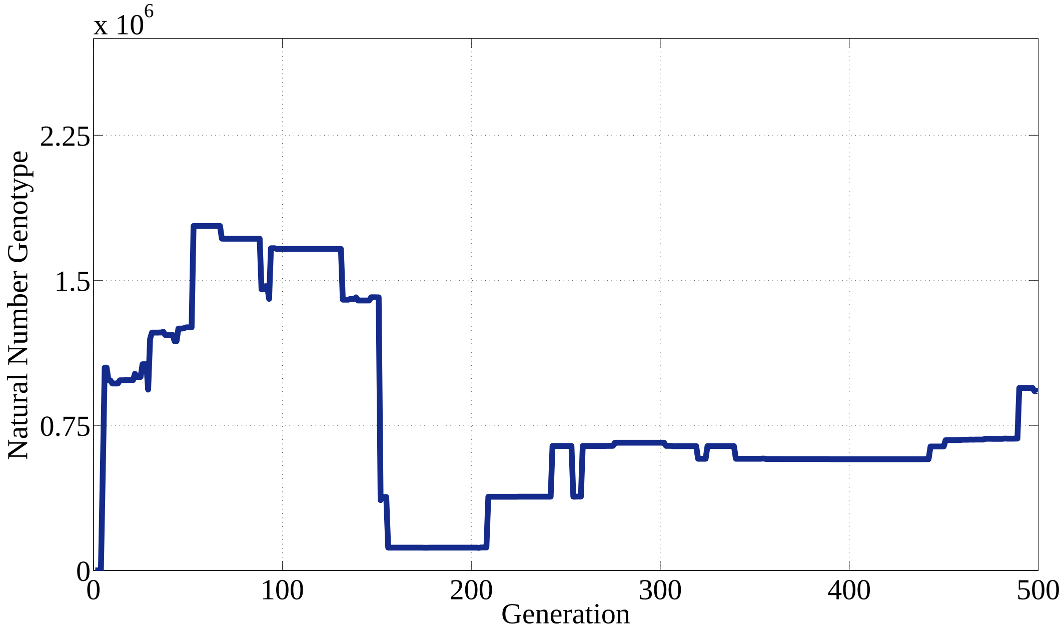


Figure 7.8: Natural number genotypes per generation.

7.3.5 Morphology

Different natural species have common physiological traits that enable basic functions. For instance, locomotion utilizes flagella, fins, wings, legs or other kinds of limbs, and energy production utilizes a metabolic system. Similarly, the recognition of a grammar by a finite-state automaton requires that a common architecture be present, regardless of species. Selective evolutionary generation serves to refine this morphological search. For the two species listed in Table 7.2, the common species morphology is associated with a symmetry transformation, where the zeros and ones of the state transitions are switched. That is, the set of unlabeled graphs of all finite-state automata in the species that recognizes a grammar with an even number of zeros is equal to the set of unlabeled graphs of all finite-state automata in the species that recognizes a grammar with an even number of ones. Again, this common morphology is easy to predict and identify due to the simplicity of the chosen grammars.

Table 7.2: Resilient Speciation Caused by Selective Evolutionary Generation

Set Number (Species Number)	Generation Range	Set Members	Set Cardinality	Common Members
1 (1)	67–150	1395719, 1395847, 1399815, 1403895, 1403911, 1412087, 1412103, 1412215, 1412231, 1453047, 1469431, 1661959, 1666039, 1666055, 1715127, 1715191, 1715383, 1780919	18	-
2 (2)	157–402	116731, 117243, 117755, 118779, 118781, 380925, 380926, 380927, 380930, 381438, 381442, 381694, 574716, 574720, 575744, 576000, 576008, 577542, 577544, 578056, 642052, 642564, 642568, 643078, 643080, 643582, 643836, 643838, 644092, 644100, 660220, 660476	32	-
3 (2)	440–442	575211	1	-

CHAPTER 8

CONCLUSIONS

This dissertation has proposed a novel on-line behavior design strategy by demonstrating and utilizing the fact that the characteristics of resilience and opportunism are guaranteed by rational behavior, the use of which is desirable because it can lead to a search that trades off prior information for search effort savings as quickly as possible. The ratio of the stationary probability of the optimizer of a fitness (or reward) function to any other element's stationary probability is given by

$$\frac{\pi_I}{\pi_j} = \left(\frac{F\left(x_I\right)}{F\left(x_j\right)} \right)^N, \ 1 \leq j \leq n, \tag{8.1}$$

where $F(x_I) > F(x_i)$ for all i implies that cell x_I is the most likely. In the limit as N approaches ∞, π_I approaches 1, and standard optimization is recovered. The canonical genetic algorithm with fitness proportional selection and the (1+1) evolutionary strategy are particular cases of the proposed scheme.

Although rational behavior suggests dynamic transitions that are based on global knowledge, this dissertation proves that rationality may be achieved through a sequence of local transitions,

$$\mathcal{X}(t+1) = Select(\mathcal{X}(t), G(\mathcal{X}(t), \mathcal{R}(t)), N), \tag{8.2}$$

that require limited knowledge of the reward function. Thus, each step of the proposed scheme is also computationally inexpensive. The resultant process is Barker's algorithm, which is a Markov Chain Monte Carlo method.

Resilience and opportunism are achieved at the expense of the mean hitting time to the optimizer, and the trade-off is managed through the level of selectivity N. The resilience of a SEGS is a conserved quantity, and any improvement to the resilience of a particular element decreases the resilience of other elements.

Multi-selective generation is an extension of the viable SEGS technique; however, the conditions for this extended scheme to behave rationally are highly restrictive. Since the technique is a generalization of the canonical genetic algorithm with fitness proportional selection, it is unlikely that a typical application of this genetic algorithm behaves rationally. Multi-selective generation can find a fit candidate optimizer faster than the original technique that it extends, but exceptions do exist.

The proposed technique has been successfully utilized without significant computational effort to 1) evolve gaits for flapping the wings of a flying vehicle in a way that is resilient with respect to changes in flight conditions, 2) evolve a disturbance rejection controller for a xerographic process in a way that is resilient with respect to internal parameter variations, and 3) evolve finite-state automata that infer a regular grammar in a way that is resilient with respect to changes in the grammar. Simulation results from the former application are used to benchmark the technique. Simulation results from the latter application demonstrate the occurrence of speciation, with evolved finite-state automata exhibiting specific physiology and morphology that enable grammar inference.

8.1 Extensions and Future Work

The results in this dissertation may be extended to the case of multi-objective (Pareto) optimization through a suitable definition of the fitness function. Future work includes understanding the origins of the Lyapunov function that characterizes a

SEGS, investigating the implications of time-reversibility, using measures of resilience and opportunism to compare evolutionary computation strategies, and exploring parallel computing implementations of the theory in this dissertation. It is also believed that a conservation law for resilience and mean hitting time exists, and preliminary efforts toward formulating this law need to be advanced. Work on a control-theoretic interpretation of the theory presented in this dissertation should continue too, since this approach holds some promise for discovering conservation laws.

Some open questions to be addressed are the following.

- Are the SEGS results invariant with respect to diffeomorphisms of cell parameters?
- What effect does incomplete knowledge of fitness (e.g., approximations or errors) have on the SEGS results?
- If the evolution of a SEGS is observed, is it possible to separately infer the fitness of the cells and the level of selectivity of the system?
- What is the level of selectivity in natural systems? Is this value the same for all natural systems or does it differ? Why?
- What happens to the SEGS results when the level of selectivity is a function of time?
- How are the SEGS results affected by limitations in the supply of a resource?

Although the results in this dissertation touch upon several open problems in artificial life, complex systems, artificial intelligence, and robotics, the practical application of selective evolutionary generation to these areas must be demonstrated in the future.

APPENDICES

APPENDIX A

PROOFS FOR CHAPTER 4

Theorem 4.1.

Proof. To show that (4.2) implies Markov chain rational behavior, consider the ratio of any π_i to π_j, $i \neq j$, where each satisfies (4.2). Equation (4.1) follows immediately.

To show that Markov chain rational behavior implies (4.2), we begin with

$$\sum_{k=1}^{n} \pi_k = 1.$$

Dividing both sides of the equation by π_i, we obtain

$$\sum_{k=1}^{n} \frac{\pi_k}{\pi_i} = \frac{1}{\pi_i},\ 1 \leq i \leq n,$$

which, using (4.1), yields

$$\sum_{k=1}^{n} \left(\frac{F(x_k)}{F(x_i)} \right)^N = \frac{1}{\pi_i},\ 1 \leq i \leq n.$$

Multiplying by $F(x_i)^N$ and solving for π_i yields (4.2), which completes the proof. □

Theorem 4.2.

Proof. We use the method of Karush-Kuhn-Tucker (KKT) multipliers to solve the optimization problem

$$\min_{\pi_1,\ldots,\pi_n} \Phi(\boldsymbol{\pi}) = -\sum_{i=1}^{n} F(x_i)^N \ln(\pi_i),$$

subject to

$$\sum_{i=1}^{n} \pi_i - 1 = 0,$$

$$-\pi_i < 0,\ 1 \leq i \leq n.$$

Let $L(\pi_1, \ldots, \pi_n, \lambda, \mu_1, \ldots, \mu_n) =$

$$-\sum_{i=1}^{n} F(x_i)^N \ln(\pi_i) + \lambda \left(\sum_{i=1}^{n} \pi_i - 1 \right) - \sum_{i=1}^{n} \mu_i \pi_i.$$

The KKT necessary conditions for optimality are

$$\frac{-F(x_i)^N}{\pi_i} + \lambda - \mu_i = 0,\ 1 \leq i \leq n,$$

$$\sum_{i=1}^{n} \pi_i - 1 = 0,$$

$$-\pi_i < 0,\ 1 \leq i \leq n,$$

$$\lambda \geq 0,$$

$$\mu_i \geq 0,\ 1 \leq i \leq n,$$

$$\lambda \left(\sum_{i=1}^{n} \pi_i - 1 \right) = 0,$$

$$\mu_i \pi_i = 0,\ 1 \leq i \leq n.$$

The first necessary condition becomes

$$-F(x_i)^N + \lambda\pi_i - \mu_i\pi_i = 0,\ 1 \le i \le n.$$

Since $\mu_i\pi_i = 0$ for all i, we obtain

$$-F(x_i)^N + \lambda\pi_i = 0,\ 1 \le i \le n.$$

Next, the constraint $\pi_i > 0$ for all i and the positive nature of $F(x_i)^N$ imply that $\lambda \neq 0$. Therefore,

$$\pi_i = \frac{F(x_i)^N}{\lambda},\ 1 \le i \le n.$$

$$\sum_{i=1}^{n} \pi_i = \sum_{i=1}^{n} \frac{F(x_i)^N}{\lambda},\ 1 \le i \le n.$$

Since $\sum\limits_{i=1}^{n} \pi_i = 1$, we find that

$$\lambda = \sum_{i=1}^{n} F(x_i)^N,$$

and hence,

$$\pi_i = \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N},\ 1 \le i \le n.$$

Thus, the stationary distribution in (4.2) satisfies the first order necessary conditions for optimality.

Moreover, we have

$$\frac{\partial^2 \Phi(\boldsymbol{\pi})}{\partial \pi_j \partial \pi_i} = 0 \text{ for } j \neq i,$$

$$\frac{\partial^2 \Phi(\boldsymbol{\pi})}{\partial \pi_i^2} = \frac{F(x_i)^N}{\pi_i^2} > 0.$$

Hence, the optimization problem has a strictly convex cost function and linear constraints. Thus, the solution of the first order necessary conditions is the global optimizer, which completes the proof. □

Theorem 4.3.

Proof. Similar to Theorem 4.2, we can use the method of Karush-Kuhn-Tucker (KKT) multipliers to solve the following optimization problem for arbitrary $\boldsymbol{\varphi} \in \mathbb{D}_n$:

$$\min_{\boldsymbol{\pi} \in \mathbb{D}_n} -\sum_{i=1}^{n} \varphi_i \ln(\pi_i),$$

which is equivalent to

$$\min_{\pi_1,\ldots,\pi_n} \Phi(\boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln(\pi_i),$$

subject to

$$\sum_{i=1}^{n} \pi_i - 1 = 0,$$

$$-\pi_i < 0, \ 1 \leq i \leq n.$$

This is a scaled version of Theorem 4.2, and therefore the remainder of the proof is omitted. □

Theorem 4.4.

Proof. We use the method of Karush-Kuhn-Tucker (KKT) multipliers to solve the optimization problem

$$\max_{\varphi_1,\ldots,\varphi_n} H(\boldsymbol{\varphi}) = -\sum_{i=1}^{n} \varphi_i \ln(\varphi_i),$$

subject to

$$\sum_{i=1}^{n} \varphi_i - 1 = 0,$$

$$-\varphi_i < 0, \ 1 \leq i \leq n,$$

$$\mathrm{E}\left[y(x)\right] - \sum_{i=1}^{n} \varphi_i y(x_i) = 0.$$

Let $L(\varphi_1, \ldots, \varphi_n, \lambda_1, \lambda_2, \mu_1, \ldots, \mu_n) =$

$$-\sum_{i=1}^{n} \varphi_i \ln(\varphi_i) + \lambda_1 \left(\sum_{i=1}^{n} \varphi_i - 1 \right) + \lambda_2 \left(\mathrm{E}\left[y(x) \right] - \sum_{i=1}^{n} \varphi_i y(x_i) \right) - \sum_{i=1}^{n} \mu_i \varphi_i .$$

The KKT necessary conditions for optimality are

$$-\ln \varphi_i - 1 + \lambda_1 - \lambda_2 y(x_i) - \mu_i = 0, \; 1 \leq i \leq n,$$

$$\sum_{i=1}^{n} \varphi_i - 1 = 0,$$

$$-\varphi_i < 0, \; 1 \leq i \leq n,$$

$$\lambda_1 \geq 0,$$

$$\lambda_2 \geq 0,$$

$$\mu_i \geq 0, \; 1 \leq i \leq n,$$

$$\lambda_1 \left(\sum_{i=1}^{n} \varphi_i - 1 \right) = 0,$$

$$\lambda_2 \left(\mathrm{E}\left[y(x) \right] - \sum_{i=1}^{n} \varphi_i y(x_i) \right) = 0,$$

$$\mu_i \varphi_i = 0, \; 1 \leq i \leq n.$$

The first necessary condition becomes

$$-\ln \varphi_i = \lambda_1 - \lambda_2 y(x_i) - \mu_i - 1, \; 1 \leq i \leq n,$$

$$\varphi_i = e^{(\lambda_1 - \lambda_2 y(x_i) - \mu_i - 1)}, \; 1 \leq i \leq n.$$

Since $\varphi_i \neq 0$ for all i, $\mu_i = 0$ for all i. We obtain

$$\varphi_i = e^{\lambda_1 - 1} \cdot e^{-\lambda_2 y(x_i)}, \; 1 \leq i \leq n,$$

or equivalently,

$$\varphi_i = \alpha e^{-\beta y(x_i)},\ 1 \leq i \leq n.$$

$\square$

Theorem 4.5.

Proof. We prove both parts of this theorem directly. Consider that

$$\begin{aligned}\rho_{ij}\Big|_{N=0} &= \frac{-N\pi_i\pi_j}{F(x_j)}\Bigg|_{N=0}, \\ &= \frac{-N}{F(x_j)} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N} \frac{F\left(x_j\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N}\Bigg|_{N=0}.\end{aligned}$$

By substitution, $\rho_{ij}\Big|_{N=0}$ is 0. Similarly,

$$\begin{aligned}\rho_{ii}\Big|_{N=0} &= \frac{N\pi_i\left(1-\pi_i\right)}{F(x_i)}\Bigg|_{N=0}, \\ &= \frac{N}{F(x_i)} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N} \left(1 - \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N}\right)\Bigg|_{N=0}.\end{aligned}$$

By substitution, $\rho_{ii}\Big|_{N=0}$ is also 0.

For the second part of the theorem, we need the following lemma.

Lemma A.1. *Let* $0 < \alpha < 1$. *Then* $\lim\limits_{N\to\infty} N\alpha^N = 0$.

Proof of Lemma A.1.

$$\begin{aligned}\lim_{N\to\infty} N\alpha^N &= \lim_{N\to\infty} \frac{N}{\alpha^{-N}} = \lim_{N\to\infty} \frac{1}{-(\alpha)^{-N}\ln\alpha} \text{ (by L'Hôpital's rule)}, \\ &= \lim_{N\to\infty} \frac{-\alpha^N}{\ln\alpha} = 0.\end{aligned}$$

$\square$

Let I be the index for which $F(x_i)$ is maximized, and assume that I is unique. Then,

$$\lim_{N\to\infty} \frac{F\left(x_j\right)^N}{F\left(x_I\right)^N} = 0,\ \forall j \neq I,\ \text{and}$$

$$\lim_{N\to\infty} \sum_{k=1}^{n} \frac{F\left(x_k\right)^N}{F\left(x_I\right)^N} = 1.$$

Consider that

$$\begin{aligned}\lim_{N\to\infty} \rho_{ij} &= \lim_{N\to 0} \frac{-N\pi_i\pi_j}{F(x_j)},\\ &= \lim_{N\to\infty} \frac{-N}{F(x_j)} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N} \frac{F\left(x_j\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N},\\ &= \lim_{N\to\infty} \frac{-N}{F(x_j)} \frac{\frac{F(x_i)^N}{F(x_I)^N}}{\sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N}} \frac{\frac{F(x_j)^N}{F(x_I)^N}}{\sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N}}.\end{aligned}$$

Now for all $i \neq j$, where $i \neq I$ and $j \neq I$, the application of Lemma A.1 with $\alpha = \frac{F(x_i)}{F(x_I)}$ implies that $\lim\limits_{N\to\infty} \rho_{ij} = 0$.

If $i = I \neq j$, then the application of Lemma A.1 with $\alpha = \frac{F(x_j)}{F(x_I)}$ implies that $\lim\limits_{N\to\infty} \rho_{ij} = 0$.

Lastly, if $i \neq j = I$, then the application of Lemma A.1 with $\alpha = \frac{F(x_i)}{F(x_I)}$ implies that $\lim\limits_{N\to\infty} \rho_{ij} = 0$.

Thus, for all i and j, $\lim\limits_{N\to\infty} \rho_{ij} = 0$.

Similarly,

$$\begin{aligned}\lim_{N\to\infty} \rho_{ii} &= \lim_{N\to 0} \frac{N\pi_i\left(1-\pi_i\right)}{F(x_i)},\\ &= \lim_{N\to\infty} \frac{N}{F(x_i)} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N} \left(1 - \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{n} F\left(x_k\right)^N}\right),\end{aligned}$$

$$= \lim_{N\to\infty} \frac{N}{F(x_i)} \frac{\frac{F(x_i)^N}{F(x_I)^N} \sum\limits_{\substack{k=1\\k\neq i}}^{n} \frac{F(x_k)^N}{F(x_I)^N}}{\sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N} \sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N}}.$$

If $i \neq I$, then the application of Lemma A.1 with $\alpha = \frac{F(x_i)}{F(x_I)}$ implies that $\lim\limits_{N\to\infty} \rho_{ii} = 0$.

If $i = I$, then we have

$$\lim_{N\to\infty} \rho_{ii} = \lim_{N\to\infty} \frac{N}{F(x_I)} \frac{\frac{F(x_I)^N}{F(x_I)^N} \sum\limits_{\substack{k=1\\k\neq I}}^{n} \frac{F(x_k)^N}{F(x_I)^N}}{\sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N} \sum\limits_{k=1}^{n} \frac{F(x_k)^N}{F(x_I)^N}}.$$

The application of Lemma A.1 with $\alpha = \frac{F(x_k)}{F(x_I)}$ a total of $n-1$ times implies that $\lim\limits_{N\to\infty} \rho_{ii} = 0$.

Thus, for all i, $\lim\limits_{N\to\infty} \rho_{ii} = 0$. This completes the proof. □

Theorem 4.6.

Proof. To show that rational behavior implies that the time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ is resilient and opportunistic, consider (4.23) and (4.24), which hold because the stationary distribution $\boldsymbol{\pi}$ has the closed form expression (4.2). By Definition 4.1, $\pi_i > 0\ \forall i$ since the Markov chain is ergodic, $N > 0$ since the Markov chain is selective, and $F(x_i) > 0\ \forall i$ since the fitness function is positive. Hence, $\rho_{ij} \neq 0\ \forall i$ and j, and $(X, \mathbf{P})$ is resilient and opportunistic. This completes the proof. □

Theorem 4.7.

Proof. To show that ergodicity is a necessary condition for the time-homogeneous, irreducible, ergodic Markov chain $(X, \mathbf{P})$ to be resilient and opportunistic, suppose that the chain is not ergodic. Then the chain is either not positive recurrent (i.e., it is null recurrent or transient) or it is periodic. If the chain is not positive recurrent, then there exists a state, x_i, with zero stationary probability. Suppose now that the fitness function is perturbed such that the fitness of this state, $F(x_i)$, becomes the optimal fitness value. Since the stationary probability of x_i is zero, state x_i is never visited, and therefore never considered as the optimizer. We have $\rho_{ii} = \partial\pi_i/\partial F(x_i) = 0$, and hence $(X, \mathbf{P})$ is not resilient or opportunistic. If the chain is periodic, then the stationary probability distribution does not exist, and resilience and opportunism are not defined. This completes the proof. $\square$

Lemma 4.1.

Proof. We prove the claim directly. Using (4.30), we have

$$\begin{aligned}\sum_{j=1}^{n}\gamma_{ij} &= \sum_{j=1}^{n}\sum_{k=1}^{m}\delta_{ijk}p_k,\\ &= \sum_{k=1}^{m}\sum_{j=1}^{n}\delta_{ijk}p_k,\\ &= \sum_{k=1}^{m}p_k\sum_{j=1}^{n}\delta_{ijk}.\end{aligned}$$

Now, $\sum\limits_{j=1}^{n}\delta_{ijk} = 1$ because cell x_i and resource r_k generate a unique cell $G(x_i, r_k)$. Therefore,

$$\sum_{j=1}^{n}\gamma_{ij} = \sum_{k=1}^{m}p_k\cdot 1 = \sum_{k=1}^{m}p_k = 1.$$

This completes the proof. $\square$

Theorem 4.8.

Proof. We prove the claim directly. Using (4.34), we have

$$\begin{aligned}\sum_{j=1}^{n}P_{ij} &= \sum_{\substack{j=1\\ j\neq i}}^{n}P_{ij} + P_{ii},\\ =&\sum_{\substack{j=1\\ j\neq i}}^{n}\frac{1}{1+\left(\frac{F(x_i)}{F(x_j)}\right)^{N}}\gamma_{ij} + \gamma_{ii} + \sum_{\substack{j=1\\ j\neq i}}^{n}\frac{1}{1+\left(\frac{F(x_j)}{F(x_i)}\right)^{N}}\gamma_{ij},\end{aligned}$$

$$
\begin{aligned}
&= \sum_{\substack{j=1\\ j\neq i}}^{n} \left(\frac{1}{1+\left(\frac{F(x_i)}{F(x_j)}\right)^N} + \frac{1}{1+\left(\frac{F(x_j)}{F(x_i)}\right)^N} \right) \gamma_{ij} + \gamma_{ii},\\
&= \sum_{\substack{j=1\\ j\neq i}}^{n} \gamma_{ij} + \gamma_{ii},\\
&= \sum_{j=1}^{n} \gamma_{ij} = 1.
\end{aligned}
$$

This completes the proof. □

Theorem 4.9.

Proof. This is a direct proof. We begin by noting that

$$
\begin{aligned}
\lim_{N\to\infty} P_{ij} &= \lim_{N\to\infty} \frac{1}{1+\left(\frac{F(x_i)}{F(x_j)}\right)^N} \gamma_{ij},\\
&= \begin{cases} \gamma_{ij}, & \text{if } F(x_i) < F(x_j),\\ 0, & \text{if } F(x_i) > F(x_j), \end{cases}
\end{aligned}
$$

and

$$
\begin{aligned}
\lim_{N\to\infty} P_{ii} &= \lim_{N\to\infty} \left(\gamma_{ii} + \sum_{\substack{j=1\\ j\neq i}}^{n} \frac{1}{1+\left(\frac{F(x_j)}{F(x_i)}\right)^N} \gamma_{ij} \right),\\
&= \gamma_{ii} + \sum_{\substack{j=1\\ j\neq i\\ F(x_i)>F(x_j)}}^{n} \gamma_{ij},\\
&= 1 - \sum_{\substack{j=1\\ j\neq i\\ F(x_i)<F(x_j)}}^{n} \gamma_{ij}.
\end{aligned}
$$

Without loss of generality, assume that the cells of the SEGS are ordered according to decreasing fitness value, so that the index $I = 1$. The matrix $\lim_{N\to\infty} \mathbf{P}$ is therefore a lower triangular matrix. Furthermore, $\lim_{N\to\infty} P_{11} = 1$.

Consider the row vector $\mathbf{v} = \begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix}$. The product of this row vector with the lower triangular matrix $\lim_{N\to\infty} \mathbf{P}$ is the first row of $\lim_{N\to\infty} \mathbf{P} = \begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix} = \mathbf{v}$.

Therefore, the row vector $\mathbf{v} = \begin{bmatrix}1 & 0 & \dots & 0\end{bmatrix}$ is a left eigenvector of $\lim_{N\to\infty} \mathbf{P}$, with corresponding eigenvalue 1 (i.e., $\mathbf{v} \lim_{N\to\infty} \mathbf{P} = \mathbf{v}$). Hence, $\lim_{N\to\infty} \boldsymbol{\pi} = \mathbf{v}$, and the proof is complete. □

Theorem 4.10.

Proof. We directly show that the row vector $\boldsymbol{\pi} = \begin{bmatrix}\pi_1 & \pi_2 & \dots & \pi_n\end{bmatrix}$, where π_i satisfies (4.2), is a left eigenvector of $\mathbf{P}$, the matrix of transition probabilities for Γ, with corresponding eigenvalue 1. If the matrix of generation probabilities, $\boldsymbol{\gamma}$, is symmetric, then

$$\gamma_{ij} = \gamma_{ji},\ 1 \le i \le n,\ 1 \le j \le n,$$

or equivalently,

$$\sum_{k=1}^{m} \delta_{ijk} p_k = \sum_{k=1}^{m} \delta_{jik} p_k.$$

Consider the row vector $\mathbf{v} = \boldsymbol{\pi}\mathbf{P}$. Then

$$\begin{aligned} v_j &= \sum_{i=1}^{n} \pi_i P_{ij}, \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j P_{jj}, \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j \left(1 - \sum_{\substack{i=1 \\ i\neq j}}^{n} P_{ji} \right), \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j - \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_j P_{ji}. \end{aligned}$$

From (4.2), (4.30), and (4.34), v_j becomes

$$\begin{aligned}&\sum_{\substack{i=1\\i\neq j}}^{n}\left(\frac{F\left(x_i\right)^N}{\sum\limits_{a=1}^{n}F\left(x_a\right)^N}\frac{F\left(x_j\right)^N}{F\left(x_i\right)^N+F\left(x_j\right)^N}\sum_{k=1}^{m}\delta_{ijk}p_k\right)\\&+\pi_j\\&-\sum_{\substack{i=1\\i\neq j}}^{n}\left(\frac{F\left(x_j\right)^N}{\sum\limits_{a=1}^{n}F\left(x_a\right)^N}\frac{F\left(x_i\right)^N}{F\left(x_i\right)^N+F\left(x_j\right)^N}\sum_{k=1}^{m}\delta_{jik}p_k\right).\end{aligned}$$

This reduces to π_j because $\boldsymbol{\gamma}$ is symmetric. Hence, $\boldsymbol{\pi}=\boldsymbol{\pi}\mathbf{P}$. □

Theorem 4.11.

Proof. We directly show that $\pi_i P_{ij}=\pi_j P_{ji}$ for all i and j. If the matrix of generation probabilities, $\boldsymbol{\gamma}$, is symmetric, then

$$\gamma_{ij}=\gamma_{ji},\ 1\leq i\leq n,\ 1\leq j\leq n,$$

or equivalently,

$$\sum_{k=1}^{m}\delta_{ijk}p_k=\sum_{k=1}^{m}\delta_{jik}p_k,\ 1\leq i\leq n,\ 1\leq j\leq n.$$

Consider $\pi_i P_{ij}$. Using (4.2), (4.30), and (4.34), we obtain

$$\begin{aligned}\pi_i P_{ij}&=\frac{F\left(x_i\right)^N}{\sum\limits_{a=1}^{n}F\left(x_a\right)^N}\frac{F\left(x_j\right)^N}{F\left(x_i\right)^N+F\left(x_j\right)^N}\sum_{k=1}^{m}\delta_{ijk}p_k,\\&=\frac{F\left(x_j\right)^N}{\sum\limits_{a=1}^{n}F\left(x_a\right)^N}\frac{F\left(x_i\right)^N}{F\left(x_i\right)^N+F\left(x_j\right)^N}\sum_{k=1}^{m}\delta_{jik}p_k,\\&=\pi_j P_{ji},\ 1\leq i\leq n,\ 1\leq j\leq n,\end{aligned}$$

where the second equation uses the symmetry of $\boldsymbol{\gamma}$. Hence, the Markov chain representing the stochastic dynamics of the ergodic SEGS is time-reversible. □

Theorem 4.12.

Proof. We use Lyapunov's Method and the LaSalle Invariance Principle [184, 185] to directly prove this theorem.

For the ergodic SEGS $\Gamma = (X, R, P, G, F)$ with a symmetric matrix of generation probabilities, $\boldsymbol{\gamma}$, consider the discrete-time dynamic system described by

$$\mathbf{p}(t+1) = \mathbf{p}(t)\mathbf{P},$$

where $\mathbf{P}$ is the matrix of transition probabilities for Γ, and $\mathbf{p}(t)$ is an n-dimensional row vector at time t. Here, $\mathbf{p}(t)$ is the ergodic probability distribution over the states at time t, and therefore the components of $\mathbf{p}$ satisfy $p_i(t) > 0,\ 1 \leq i \leq n$, and $\sum\limits_{i=1}^{n} p_i(t) = 1$. Since the SEGS is ergodic and irreducible, a unique equilibrium stationary distribution for these dynamics exists, $\lim\limits_{t\to\infty} \mathbf{p}(t) = \boldsymbol{\pi}$, with components π_i satisfying (4.2).

Let us define $\mathbf{q}(t) = \mathbf{p}(t) - \boldsymbol{\pi}$, so that the transformed discrete-time dynamic system,

$$\mathbf{q}(t+1) = (\mathbf{q}(t) + \boldsymbol{\pi})\,\mathbf{P} - \boldsymbol{\pi},$$

has an equilibrium at the origin. The function

$$V(\mathbf{p}(t)) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{p_i(t)}{\varphi_i}\right),$$

where φ_i satisfies (4.6), may be rewritten as

$$V(\mathbf{q}(t) + \boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{q_i(t) + \pi_i}{\varphi_i}\right).$$

We first check the value of this transformed candidate Lyapunov equation at the origin of the transformed system. We have

$$V(\mathbf{0}+\boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{\pi_i}{\varphi_i}\right) = -\sum_{i=1}^{n} \varphi_i \ln 1 = 0,$$

because $\boldsymbol{\pi} = \boldsymbol{\varphi}$.

Next, we have to show that $\forall \mathbf{q}(t) \neq \mathbf{0}$, $V(\mathbf{q}(t)+\boldsymbol{\pi}) > 0$. But this follows directly from (the second equivalent restatement of) Theorem 4.3. This is because $\forall \mathbf{q}(t) \neq \mathbf{0}$,

$$V(\mathbf{q}(t)+\boldsymbol{\pi}) = -\sum_{i=1}^{n} \varphi_i \ln\left(\frac{p_i(t)}{\varphi_i}\right),$$

which is always positive according to the theorem.

Now consider $\Delta V = V(\mathbf{q}(t+1)+\boldsymbol{\pi}) - V(\mathbf{q}(t)+\boldsymbol{\pi})$. In the equations that follow, we assume, without loss of generality, that the fitness value of each cell of the SEGS is greater than or equal to one. (After all, if there exists an i such that $0 < F(x_i) < 1$, then it is possible to find a $K \in \mathbb{R}^+$ to scale all the fitness values upward, so that for all i, $KF(x_i) \geq 1$. Define the new fitnesses $F'(x_i) = KF(x_i),\ 1 \leq i \leq n$ and observe that the Markov chain representation of the SEGS is unchanged).

$$\begin{aligned}
\Delta V &= V(\mathbf{p}(t+1)) - V(\mathbf{p}(t)),\\
&= -\sum_{j=1}^{n} \varphi_j \ln\left(\frac{p_j(t+1)}{\varphi_j}\right) + \sum_{j=1}^{n} \varphi_j \ln\left(\frac{p_j(t)}{\varphi_j}\right),\\
&= -\sum_{j=1}^{n} \varphi_j \ln\left(\frac{p_j(t+1)}{p_j(t)}\right),\\
&= -\sum_{j=1}^{n} \varphi_j \ln\left(\frac{\sum_{i=1}^{n} p_i(t)P_{ij}}{p_j(t)}\right),\\
&= -\sum_{j=1}^{n} \varphi_j \ln\left(\frac{1}{p_j(t)} \sum_{i=1}^{n} \frac{p_i(t)F(x_j)^N}{F(x_i)^N + F(x_j)^N}\right),\\
&= -\sum_{j=1}^{n} \varphi_j \ln\left(\frac{F(x_j)^N}{p_j(t)} \sum_{i=1}^{n} \frac{p_i(t)}{F(x_i)^N + F(x_j)^N}\right).
\end{aligned}$$

Now because we have assumed, without loss of generality, that all fitnesses are greater than or equal to one, we have

$$\begin{aligned}
&\frac{F(x_j)^N}{p_j(t)} \sum_{i=1}^{n} \frac{p_i(t)}{F(x_i)^N + F(x_j)^N} \\
\geq &\sum_{i=1}^{n} \frac{p_i(t)}{F(x_i)^N + F(x_j)^N}, \\
\geq &\sum_{i=1}^{n} p_i(t), \\
\geq &1.
\end{aligned}$$

Therefore, we obtain

$$\begin{aligned}
&\Delta V \leq -\sum_{j=1}^{n} \varphi_j \ln 1, \text{ or} \\
&\Delta V \leq 0.
\end{aligned}$$

That is, ΔV is negative semi-definite, as required by Lyapunov's method.

To apply LaSalle's Invariance Principle, we have to find $Q = \{\mathbf{q}(t) | \Delta V = 0\}$. Note that

$$\begin{aligned}
\Delta V &= 0, \\
&= -\sum_{j=1}^{n} \varphi_j \ln 1, \\
&= -\sum_{j=1}^{n} \varphi_j \ln \left(\sum_{i=1}^{n} P_{ji} \right),
\end{aligned}$$

which can be rewritten with Bayes' Rule as

$$\Delta V = -\sum_{j=1}^{n} \varphi_j \ln \left(\sum_{i=1}^{n} \frac{\pi_i}{\pi_j} P_{ij} \right).$$

We had previously shown that

$$\Delta V = -\sum_{j=1}^{n} \varphi_j \ln \left(\frac{\sum\limits_{i=1}^{n} p_i(t) P_{ij}}{p_j(t)} \right).$$

Thus, $\Delta V = 0$ implies that $\pi_i = p_i(t),\ 1 \leq i \leq n$. But from the definition of $\mathbf{q}(t)$,

$$\pi_i = p_i(t) - q_i(t),$$

and we must have that $\Delta V = 0$ implies that $q_i(t) = 0,\ 1 \leq i \leq n$. Therefore, the only solution of the transformed discrete-time dynamic system that can stay identically in Q is the trivial solution $\mathbf{q}(t) \equiv 0$. Hence, the origin is an asymptotically stable equilibrium for the transformed discrete-time dynamic system, and therefore, the function

$$V(\mathbf{p}(t)) = -\sum_{i=1}^{n} \varphi_i \ln \left(\frac{p_i(t)}{\varphi_i} \right),$$

is a Lyapunov function for the original system with the set of vectors $\mathbf{p}$ with components $p_i(t) > 0,\ 1 \leq i \leq n$, and $\sum\limits_{i=1}^{n} p_i(t) = 1$ forming an invariant manifold. Moreover, since the Lyapunov function is radially unbounded, the equilibrium is globally asymptotically stable, as claimed. □

Theorem 4.13.

Proof. We first prove directly that σ_{iI} converges to a constant value for each i as N approaches ∞, before inductively showing that the value of σ_{iI} does indeed decrease with increasing N.

We begin by noting that

$$\lim_{N \to \infty} P_{ij} = \lim_{N \to \infty} \frac{1}{1 + \left(\frac{F(x_i)}{F(x_j)} \right)^N} \gamma_{ij},$$

$$= \begin{cases} \gamma_{ij}, & \text{if } F(x_i) < F(x_j), \\ 0, & \text{if } F(x_i) > F(x_j), \end{cases}$$

and

$$\begin{aligned} \lim_{N\to\infty} P_{ii} &= \lim_{N\to\infty} \left(\gamma_{ii} + \sum_{\substack{j=1 \\ j\neq i}}^{n} \frac{1}{1 + \left(\frac{F(x_j)}{F(x_i)}\right)^N} \gamma_{ij} \right), \\ &= \gamma_{ii} + \sum_{\substack{j=1 \\ j\neq i \\ F(x_i)>F(x_j)}}^{n} \gamma_{ij}, \\ &= 1 - \sum_{\substack{j=1 \\ j\neq i \\ F(x_i)<F(x_j)}}^{n} \gamma_{ij}. \end{aligned}$$

Without loss of generality, assume that the cells of the SEGS are ordered according to decreasing fitness value, so that the index $I = 1$. The matrix $\lim_{N\to\infty} \mathbf{P}$ is therefore a lower triangular matrix.

We seek

$$\begin{aligned} \lim_{N\to\infty} \boldsymbol{\sigma}_1 &= \lim_{N\to\infty} (\mathbf{I} - \mathbf{D}_1\mathbf{P})^{-1}\mathbf{D}_1\mathbf{1}, \\ &= (\mathbf{I} - \mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})^{-1}\mathbf{D}_1\mathbf{1}, \end{aligned}$$

where $(\mathbf{I} - \mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})^{-1}$ always exists due to the following.

(i) $\lim_{N\to\infty} \mathbf{P}$ is a lower triangular matrix with full rank. All of the lower triangular elements are non-zero.

(ii) $\mathbf{D}_1$ is a lower triangular matrix with rank $n - 1$.

(iii) $(\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})$ is a matrix with zeros in row one, and elements that are equal to $\lim_{N\to\infty} \mathbf{P}$ in all other rows. Hence, $(\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})$ has rank $n - 1$. Since this matrix is the product of lower triangular matrices, it is also lower triangular.

(iv) $(\mathbf{I}-\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})$ is a lower triangular matrix because it is the difference of lower triangular matrices. All lower triangular elements of this matrix are non-zero, with the matrix element $(\mathbf{I}-\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})_{11} = 1$. Thus, $(\mathbf{I}-\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})$ has full rank.

Since $(\mathbf{I}-\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P})$ is a lower triangular matrix with full rank, the equation

$$(\mathbf{I}-\mathbf{D}_1 \lim_{N\to\infty} \mathbf{P}) \lim_{N\to\infty} \boldsymbol{\sigma}_1 = \mathbf{D}_1\mathbf{1},$$

may be solved by the iterative process of forward substitution to obtain unique constant values of $\lim_{N\to\infty} \sigma_{i1}$ for each i. For instance,

$$\begin{aligned}
\lim_{N\to\infty} \sigma_{11} &= 0,\\
\lim_{N\to\infty} \sigma_{21} &= \frac{1}{\gamma_{21}},\\
\lim_{N\to\infty} \sigma_{31} &= \frac{1+\gamma_{32}(\lim_{N\to\infty} \sigma_{21})}{\gamma_{31}+\gamma_{32}},\\
&= \frac{1+\frac{\gamma_{32}}{\gamma_{21}}}{\gamma_{31}+\gamma_{32}},\\
\lim_{N\to\infty} \sigma_{41} &= \frac{1+\gamma_{42}(\lim_{N\to\infty} \sigma_{21})+\gamma_{43}(\lim_{N\to\infty} \sigma_{31})}{\gamma_{41}+\gamma_{42}+\gamma_{43}},\\
&= \frac{1+\frac{\gamma_{42}}{\gamma_{21}}+\gamma_{43}\left(\frac{1+\frac{\gamma_{32}}{\gamma_{21}}}{\gamma_{31}+\gamma_{32}}\right)}{\gamma_{41}+\gamma_{42}+\gamma_{43}},
\end{aligned}$$

and so on. This completes the convergence part of the proof.

We next use induction on the cell index to show that σ_{i1} is a strictly decreasing function of N. First, consider that

$$\sigma_{21} = \frac{\sum\limits_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\left(\sigma_{k1}+1\right)+P_{22}}{1-P_{22}}.$$

Hence,

$$\begin{aligned}
\lim_{N\to\infty} \sigma_{21} &= \lim_{N\to\infty} \frac{1+\sum\limits_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\sigma_{k1}}{1-P_{22}}, \\
&= \frac{\lim\limits_{N\to\infty}\left(1+\sum\limits_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\sigma_{k1}\right)}{\lim\limits_{N\to\infty}\left(1-P_{22}\right)}, \\
&= \frac{1+\lim\limits_{N\to\infty}\left(\sum\limits_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\sigma_{k1}\right)}{1-\left(1-\gamma_{21}\right)}, \\
&= \frac{1}{\gamma_{21}} + \frac{1}{\gamma_{21}} \lim_{N\to\infty}\left(\sum_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\sigma_{k1}\right).
\end{aligned}$$

Comparing this expression to the result that was calculated by forward substitution above,

$\lim\limits_{N\to\infty} \sum\limits_{\substack{k=1\\k\neq 2}}^{n} P_{2k}\sigma_{k1}$ must decrease to 0 as N increases. Therefore, σ_{21} decreases as N increases.

For the induction hypothesis, assume that for any $s-1$ where $2 \leq (s-1) \leq (n-1)$, we have that for all t where $2 \leq t \leq (s-1)$, the mean hitting time σ_{t1} decreases with N. We now show that σ_{s1} is a decreasing function of N.

Consider that

$$\sigma_{s1} = \frac{\sum\limits_{\substack{k=1\\k\neq s}}^{n} P_{sk}\left(\sigma_{k1}+1\right) + P_{ss}}{1-P_{ss}}.$$

Hence,

$$\lim_{N\to\infty} \sigma_{s1} = \lim_{N\to\infty} \frac{1+\sum\limits_{\substack{k=1\\k\neq s}}^{n} P_{sk}\sigma_{k1}}{1-P_{ss}},$$

$$= \frac{\lim\limits_{N\to\infty}\left(1+\sum\limits_{\substack{k=1\\k\neq s}}^{n} P_{sk}\sigma_{k1}\right)}{\lim\limits_{N\to\infty}(1-P_{ss})},$$

$$= \frac{1+\lim\limits_{N\to\infty}\left(\sum\limits_{\substack{k=1\\k\neq s}}^{n} P_{sk}\sigma_{k1}\right)}{\sum\limits_{k=1}^{s-1}\gamma_{sk}},$$

$$= \frac{1+\lim\limits_{N\to\infty}\left(\sum\limits_{k=1}^{s-1} P_{sk}\sigma_{k1}+\sum\limits_{k=s+1}^{n} P_{sk}\sigma_{k1}\right)}{\sum\limits_{k=1}^{s-1}\gamma_{sk}}.$$

Comparing this expression to the general result calculated by forward substitution, $\lim\limits_{N\to\infty}\sum\limits_{k=s+1}^{n} P_{sk}\sigma_{k1}$ must decrease to 0 as N increases. By the induction hypothesis, $\lim\limits_{N\to\infty}\sum\limits_{k=1}^{s-1} P_{sk}\sigma_{k1}$ decreases with increasing N. Therefore, σ_{s1} is a decreasing function of N.

Hence, for all i where $2 \le i \le n$, an increase in the level of selectivity produces a corresponding decrease in the mean hitting time to the fittest cell, σ_{i1}, with $\lim\limits_{N\to\infty}\sigma_{i1}$ approaching a unique constant value for each i. □

Theorem 4.14.

Proof.

1. We directly show that the row vector $\boldsymbol{\pi} = \begin{bmatrix}\pi_1 & \pi_2 & \dots & \pi_n\end{bmatrix}$, where π_i satisfies (4.2), is a left eigenvector of $\mathbf{P}$, the matrix of transition probabilities for Γ, with corresponding eigenvalue 1. If the descendancy matrix, $\boldsymbol{\delta}$, is symmetric, then

$$\delta_{ij} = \delta_{ji},\ 1 \le i \le n,\ 1 \le j \le n.$$

Consider the row vector $\mathbf{v} = \boldsymbol{\pi}\mathbf{P}$. Then

$$\begin{aligned} v_j &= \sum_{i=1}^{n} \pi_i P_{ij}, \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j P_{jj}, \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j \left(1 - \sum_{\substack{i=1 \\ i\neq j}}^{n} P_{ji} \right), \\ &= \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_i P_{ij} + \pi_j - \sum_{\substack{i=1 \\ i\neq j}}^{n} \pi_j P_{ji}. \end{aligned}$$

From (4.2) and (4.90), v_j becomes

$$\begin{aligned} &\sum_{\substack{i=1 \\ i\neq j}}^{n} \left(\frac{F(x_i)^N}{\sum\limits_{a=1}^{n} F(x_a)^N} \frac{F(x_j)^N}{\sum\limits_{k=1}^{m} F(G(x_i, r_k))^N + F(x_i)^N} \delta_{ij} \right) \\ &+\pi_j \\ &-\sum_{\substack{i=1 \\ i\neq j}}^{n} \left(\frac{F(x_j)^N}{\sum\limits_{a=1}^{n} F(x_a)^N} \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F(G(x_j, r_k))^N + F(x_j)^N} \delta_{ji} \right). \end{aligned}$$

This reduces to π_j because of assumptions i) and ii). Hence, $\boldsymbol{\pi} = \boldsymbol{\pi}\mathbf{P}$.

2. We directly show that $\pi_i P_{ij} = \pi_j P_{ji}$ for all i and j. If the descendancy matrix, $\boldsymbol{\delta}$, is symmetric, then

$$\delta_{ij} = \delta_{ji},\ 1 \leq i \leq n,\ 1 \leq j \leq n.$$

Consider $\pi_i P_{ij}$. Using (4.2) and (4.90), we obtain

$$\pi_i P_{ij} = \frac{F(x_i)^N}{\sum\limits_{a=1}^{n} F(x_a)^N} \frac{F(x_j)^N}{\sum\limits_{k=1}^{m} F(G(x_i, r_k))^N + F(x_i)^N} \delta_{ij},$$

$$= \frac{F\left(x_j\right)^N}{\sum\limits_{a=1}^{n} F\left(x_a\right)^N} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} \delta_{ji},$$

$$= \pi_j P_{ji}, \; 1 \le i \le n, \; 1 \le j \le n,$$

where the second equation uses assumptions i) and ii). Hence, the Markov chain representing the stochastic dynamics of the ergodic SEGS is time-reversible.

□

Theorem 4.15.

Proof. We directly show that (4.93) and (4.94) follow from the stated definition of rational behavior. Consider the stationary probability distribution

$$\boldsymbol{\pi} = \boldsymbol{\pi} \mathbf{P}.$$

Equivalently,

$$\begin{aligned} \pi_j &= \sum_{i=1}^{n} \pi_i P_{ij}, \\ &= \sum_{i=1}^{n} \pi_i \frac{F\left(x_j\right)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij}, \end{aligned}$$

where we have made use of (4.90). Substituting (4.2), we obtain

$$F(x_j)^N = \sum_{i=1}^{n} F(x_i)^N \frac{F\left(x_j\right)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij},$$

which can be simplified to

$$1 = \sum_{i=1}^{n} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij}.$$

Since $\sum\limits_{i=1}^{n} P_{ji} = 1$, we can write

$$\sum_{i=1}^{n} P_{ji} = \sum_{i=1}^{n} \frac{F\left(x_i\right)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij},$$

which, using (4.90) again, is

$$\sum_{i=1}^{n} \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} \delta_{ji} = \sum_{i=1}^{n} \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij}.$$

Now, if the Markov chain representing the stochastic dynamics of the ergodic MSGS is also time-reversible, then

$$\pi_i P_{ij} = \pi_j P_{ji}.$$

Substituting (4.2) and (4.90),

$$F(x_i)^N \frac{F(x_j)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N} \delta_{ij} = F(x_j)^N \frac{F(x_i)^N}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} \delta_{ji},$$

which simplifies to

$$\frac{\delta_{ji}}{\sum\limits_{k=1}^{m} F\left(G\left(x_j, r_k\right)\right)^N + F\left(x_j\right)^N} = \frac{\delta_{ij}}{\sum\limits_{k=1}^{m} F\left(G\left(x_i, r_k\right)\right)^N + F\left(x_i\right)^N}.$$

□

APPENDIX B

SELECTED PSEUDOCODE

Pseudocode: Convert *input*, a natural number, into a $\{1,0\}$ deterministic finite-state automaton (DFSA).

```
Require: $input \in \mathbb{N}$
Output: $[NumStates, s, A, Delta]$
// Some preliminaries.
$MaxDFSANumStates \leftarrow 25$ // Maximum number of DFSA states can be varied, as required.
$NatNumBias \leftarrow 0$
for $counter = 1$ to $MaxDFSANumStates$ do
    $DeltaNaturalNumber(counter) \leftarrow (2\,\hat{}\,counter) * (counter\,\hat{}\,(2 * counter + 1))$
    $NaturalNumberLimit(counter) \leftarrow DeltaNaturalNumber(counter) + NatNumBias$
    $NatNumBias \leftarrow NaturalNumberLimit(counter)$
end for
// Determine the number of states in the DFSA.
$NumStates \leftarrow 1$
while $input > NaturalNumberLimit(NumStates)$ do
    $NumStates \leftarrow NumStates + 1$
end while
```

```
// Determine the index of the start state.
if $NumStates > 1$ then
    $Remainder \leftarrow input - NaturalNumberLimit(NumStates - 1)$
else
    $Remainder \leftarrow input$
end if
$s \leftarrow \lceil Remainder/(2 \ \hat{} \ NumStates)/(NumStates \ \hat{} \ (2 * NumStates)) \rceil$
// Determine the cardinality of the set of accept states, $CardA$.
if $s > 1$ then
    $Remainder2 \leftarrow Remainder \bmod$
            $((s - 1) * (2 \ \hat{} \ NumStates) * (NumStates \ \hat{} \ (2 * NumStates)))$
else
    $Remainder2 \leftarrow Remainder \bmod$
            $((2 \ \hat{} \ NumStates) * (NumStates \ \hat{} \ (2 * NumStates)))$
end if
$CardA \leftarrow 0$
$i \leftarrow 0$
while $Remainder2 > 0$ do
    $Remainder2 \leftarrow Remainder2 - \binom{NumStates}{i} * (NumStates \ \hat{} \ (2 * NumStates))$
    $i \leftarrow i + 1$
end while
if $i > 0$ then
    $CardA \leftarrow i - 1$
    $Remainder2 \leftarrow Remainder2 + \binom{NumStates}{CardA} * (NumStates \ \hat{} \ (2 * NumStates))$
```

```
else
    $CardA \leftarrow NumStates$
end if
// Determine the set of accept states, $A$.
Initialize $A$.
if $CardA > 0$ then
    if $Remainder2 \bmod (NumStates \,\hat{}\, (2 * NumStates)) > 0$ then
        $Bias \leftarrow \lfloor Remainder2 / (NumStates \,\hat{}\, (2 * NumStates)) \rfloor$
    else
        if $Remainder2 > 0$ then
            $Bias \leftarrow \lfloor (Remainder2 - 1) / (NumStates \,\hat{}\, (2 * NumStates)) \rfloor$
        else
            $Bias \leftarrow NumStates - CardA$
        end if
    end if
    for $ctr = 1$ to $CardA$ do
        if $ctr + Bias <= NumStates$ then
            $A \leftarrow [A, ctr + Bias]$
        else
            $A \leftarrow [A, ctr + Bias - NumStates]$
        end if
    end for
else
    $Bias \leftarrow 0$
end if
// Determine the Delta transitions.
$Remainder3 \leftarrow Remainder2 \bmod (Bias * (NumStates \,\hat{}\, (2 * NumStates)))$
```

```
if $NumStates > 1$ then
    $DeltaString \leftarrow$ conversion of $Remainder3$ to a base of $NumStates$
    if $DeltaString.length < 2 * NumStates$ then
        $LengthToAdd \leftarrow 2 * NumStates - DeltaString.length$
        for $j = 1$ to $LengthToAdd$ do
            $DeltaString \leftarrow$ concatenate the string ‘0’ with $DeltaString$
        end for
    else if $DeltaString.length > 2 * NumStates$ then
        $LengthToSubtract \leftarrow DeltaString.length - 2 * NumStates$
        $TempString \leftarrow DeltaString$
        $DeltaString \leftarrow$ the empty string ‘’
        for $j = 1 + LengthToSubtract$ to $TempString.length$ do
            $DeltaString(j - LengthToSubtract) \leftarrow TempString(j)$
        end for
    end if
else
    $DeltaString \leftarrow$ the string ‘0’
end if
Initialize $Delta$.
// Add the transitions in matrix form.
for $k = 1$ to $DeltaString.length$ step 2 do
    if $DeltaString.length > 1$ then
        $Delta \leftarrow [Delta; (DeltaString(k))$ converted to a number, $(DeltaString(k + 1))$ converted to a number] // i.e., add a two column row.
    else
        $Delta \leftarrow [Delta; (DeltaString(k))$ converted to a number] // i.e., add a row.
    end if
```

```
end for
// Shift the state indices to start at 1 instead of 0.
if $NumStates > 1$ then
    $Delta \leftarrow Delta +$ a matrix of ones with dimensions $NumStates \times 2$
else
    $Delta \leftarrow Delta + 1$
end if
```

REFERENCES

REFERENCES

[1] Michael J. Atallah, editor. *Algorithms and Theory of Computation Handbook*. CRC Press, 1999.

[2] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3rd edition, 2009.

[3] Harold W. Kuhn and Albert W. Tucker. Nonlinear programming. In *Proceedings of the Second Berkeley Symposium of Mathematical Statistics and Probability*. University of California Press, 1951.

[4] J. E. Dennis, Jr. and Robert B. Schnabel. *Numerical Methods for Unconstrained Optimization and Nonlinear Equations*. SIAM, 1996.

[5] James M. Ortega and Werner C. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. SIAM, 2000.

[6] David G. Luenberger. *Linear and Nonlinear Programming*. Kluwer Academic Publishers, 2nd edition, 2003.

[7] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.

[8] Allan Borodin and Ran El-Yaniv. *Online Computation and Competitive Analysis*. Cambridge University Press, 1998.

[9] Amos Fiat and Gerhard J. Woeginger, editors. *Online Algorithms: The State of the Art*, volume 1442 of *Lecture Notes in Computer Science*. Springer, 1998.

[10] N. Ascheuer, Martin Grötschel, Sven O. Krumke, and Jörg Rambau. Combinatorial online optimization. In Peter Kall and Hans-Jakob Lüthi, editors, *Operations Research Proceedings 1998*, pages 21–37. Springer, 1999.

[11] Martin Grötschel, Sven O. Krumke, and Jörg Rambau, editors. *Online Optimization of Large Scale Systems*. Springer, 2001.

[12] Susanne Albers. Online algorithms: A survey. *Mathematical Programming*, 97(1–2):3–26, July 2003.

[13] Kartik B. Ariyur and Miroslav Krstić. *Real-Time Optimization by Extremum-Seeking Control*. Wiley, 2003.

[14] Pascal Van Hentenryck and Russel Bent. *Online Stochastic Combinatorial Optimization*. MIT Press, October 2006.

[15] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms*. MIT Press, 2nd edition, 2001.

[16] Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.

[17] S. Kirkpatrick, C. D. Gelatt, Jr., and M. P. Vecchi. Optimization by simulated annealing. *Science*, 220(4598):671–680, 1983.

[18] A. Corana, M. Marchesi, C. Martini, and S. Ridella. Minimizing multimodal functions of continuous variables with the 'simulated annealing' algorithm. *ACM Transactions on Mathematical Software*, 13(3):262–280, 1987.

[19] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley, 1989.

[20] Lawrence Davis, editor. *Handbook of Genetic Algorithms*. Van Nostrand Reinhold, 1991.

[21] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, 1996.

[22] Ingo Rechenberg. *Evolutionsstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. PhD thesis, Technical University of Berlin, 1971.

[23] Hans-Paul Schwefel. *Evolution and Optimum Seeking*. Wiley, 1995.

[24] Hans-Georg Beyer and Hans-Paul Schwefel. Evolution strategies: A comprehensive introduction. *Natural Computing*, 1(1):3–52, 2002.

[25] David B. Fogel. *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. Wiley, 3rd edition, 2006.

[26] Hans-Georg Beyer. *The Theory of Evolution Strategies*. Springer, 2001.

[27] J. Maynard Smith. Optimization theory in evolution. *Annual Review of Ecology and Systematics*, 9:31–56, November 1978.

[28] Ian Dempsey, Michael O'Neill, and Anthony Brabazon. *Foundations in Grammatical Evolution for Dynamic Environments*. Springer, 2009.

[29] Semyon M. Meerkov. Mathematical theory of behavior — individual and collective behavior of retardable elements. *Mathematical Biosciences*, 43(1–2):41–106, 1979.

[30] John von Neumann. *Theory of Self-Reproducing Automata*. University of Illinois Press, 1966.

[31] Moshe Sipper. Fifty years of research on self-replication: An overview. *Artificial Life*, 4(3):237–257, 1998.

[32] Robert A. Freitas, Jr. and Ralph C. Merkle. *Kinematic Self-Replicating Machines*. Landes Bioscience, 2004.

[33] Patrick D. Owens and A. Galip Ulsoy. Self-replicating machines: Prevention of degeneracy. Technical Report CGR-06-02, University of Michigan, 2006.

[34] Bernard Foing. Roadmap for robotic and human exploration of the moon and beyond. In *Proceedings of the 56th International Astronautical Congress*, number IAC-05-A5.1.01, 17–21 October 2005.

[35] James R. Wertz and Wiley J. Larson, editors. *Space Mission Analysis and Design*. Microcosm Press, 3rd edition, 1999.

[36] Robert A. Freitas, Jr. and William P. Gilbreath, editors. *Advanced Automation for Space Missions*, number N83-15348 (NASA Conference Publication CP-2255), 1982.

[37] Gregory S. Chirikjian, Yu Zhou, and Jackrit Suthakorn. Self-replicating robots for lunar development. *IEEE/ASME Transactions on Mechatronics*, 7(4), December 2002.

[38] T. Fukuda, S. Nakagawa, Y. Kawauchi, and M. Buss. Self-organizing robots based on cell structures - CEBOT. In *Proceedings of the 1988 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 145–150, 31 October-2 November 1988.

[39] Cem Ünsal, Han Kiliççöte, and Pradeep K. Khosla. A modular self-reconfigurable bipartite robotic system: Implementation and motion planning. *Autonomous Robots*, 10(1):23–40, January 2001.

[40] Andres Castano, Alberto Behar, and Peter M Will. The Conro modules for reconfigurable robots. *IEEE/ASME Transactions on Mechatronics*, 7(4):403–409, December 2002.

[41] Satoshi Murata, Eiichi Yoshida, Akiya Kamimura, Haruhisa Kurokawa, Kohji Tomita, and Shigeru Kokaji. M-TRAN: Self-reconfigurable modular robotic system. *IEEE/ASME Transactions on Mechatronics*, 7(4):431–441, December 2002.

[42] Mark Yim, Kimon Roufas, David Duff, Ying Zhang, Craig Eldershaw, and Sam Homans. Modular reconfigurable robots in space applications. *Autonomous Robots*, 14(2–3):225–237, March 2003.

[43] Morten Winkler Jørgensen, Esben Hallundbæk Østergaard, and Henrik Hautop Lund. Modular ATRON: Modules for a self-reconfigurable robot. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 2068–2073, 28 September-2 October 2004.

[44] Victor Zykov, Efstathios Mytilinaios, Bryant Adams, and Hod Lipson. Self-reproducing machines. *Nature*, 435:163–164, 12 May 2005.

[45] Seth Copen Goldstein, Jason D. Campbell, and Todd C. Mowry. Programmable matter. *IEEE Computer*, 38(6):99–101, June 2005.

[46] Saul Griffith, Dan Goldwater, and Joseph M. Jacobson. Self-replication from random parts. *Nature*, 437:636, 29 September 2005.

[47] Behnam Salemi, Mark Moll, and Wei-Min Shen. SUPERBOT: A deployable, multi-functional, and modular self-reconfigurable robotic system. In *Proceedings of the 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3636–3641, 9–15 October 2006.

[48] Kyle Gilpin, Keith Kotay, and Daniela Rus. Miche: Modular shape formation by self-dissasembly. In *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pages 2241–2247, 10–14 April 2007.

[49] Eric Klavins. Programmable self-assembly. *IEEE Control Systems Magazine*, 27(4):43–56, August 2007.

[50] Anders Lyhne Christensen, Rehan O'Grady, and Marco Dorigo. Morphology control in a multirobot system: Distributed growth of specific structures using directional self-assembly. *IEEE Robotics & Automation Magazine*, 14(4):18–25, 2007.

[51] A. Scott Howe. Self-assembling modular robotic structures: Standard units with swappable payloads. *IEEE Robotics & Automation Magazine*, 14(4):26–33, 2007.

[52] Mark Yim, Wei-Min Shen, Behnam Salemi, Daniela Rus, Mark Moll, Hod Lipson, Eric Klavins, and Gregory S. Chrikjian. Modular self-reconfigurable robot systems: Challenges and opportunities for the future. *IEEE Robotics & Automation Magazine*, 14(1):43–52, 2007.

[53] Satoshi Murata and Haruhisa Kurokawa. Self-reconfigurable robots: Shape-changing cellular robots can exceed conventional robot flexibility. *IEEE Robotics & Automation Magazine*, 14(1):71–78, 2007.

[54] Robert Fitch and Daniela Rus. Self-reconfiguring robots in the USA. *Journal of the Robotics Society of Japan*, 21(8):4–10, November 2003.

[55] Esben H. Østergaard. *Distributed Control of the ATRON Self-reconfigurable Robot*. PhD thesis, University of Southern Denmark, 18 November 2004.

[56] David Johan Christensen. *Elements of Autonomous Self-reconfigurable Robots*. PhD thesis, University of Southern Denmark, June 2008.

[57] Zack Butler, Robert Fitch, and Daniela Rus. Distributed control for unit-compressible robots: Goal-recognition, locomotion, and splitting. *IEEE/ASME Transactions on Mechatronics*, 7(4):418–430, December 2002.

[58] Jennifer E. Walter, Jennifer L. Welch, and Nancy M. Amato. Distributed reconfiguration of metamorphic robot chains. *Distributed Computing*, 17(2):171–189, August 2004.

[59] Mark Yim, Babak Shirmohammadi, Jimmy Sastra, Michael Park, Michael Dugan, and C. J. Taylor. Towards robotic self-reassembly after explosion. In *Proceedings of the 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2767–2772, 29 October-2 November 2007.

[60] David Johan Christensen. Experiments on fault-tolerant self-reconfiguration and emergent self-repair. In *Proceedings of the IEEE Symposium on Artificial Life*, pages 355–361, 1–5 April 2007.

[61] Kasper Stoy and Radhika Nagpal. Self-repair through scale-independent self-reconfiguration. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 2, pages 2062–2067, 28 September-2 October 2004.

[62] Paulina Varshavskaya, Leslie Pack Kaelbling, and Daniela Rus. Learning distributed control for modular robots. In *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2648–2653, 28 September-2 October 2004.

[63] Zack Butler, Keith Kotay, Daniela Rus, and Kohji Tomita. Cellular automata for decentralized control of self-reconfigurable robots. In *Proceedings of the 2002 IEEE International Conference on Robotics and Automation*, volume 1, pages 809–816, 11–15 May 2002.

[64] Wei-Min Shen, Peter Will, Aram Galstyan, and Cheng-Ming Chuong. Hormone-inspired self-organization and distributed control of robotic swarms. *Autonomous Robots*, 17(1):93–105, July 2004.

[65] Gregory S. Chirikjian. Parts entropy, symmetry, and the difficulty of self-replication. In *Proceedings of the 2008 ASME Dynamic Systems and Control Conference*, number DSCC2008-2280, 20–22 October 2008.

[66] Dario Floreano, Sara Mitri, Stéphane Magnenat, and Laurent Keller. Evolutionary conditions for the emergence of communication in robots. *Current Biology*, 17(6):514–519, 20 March 2007.

[67] Crawford Stanley Holling. Resilience and stability of ecological systems. *Annual Review of Ecology and Systematics*, 4:1–23, November 1973.

[68] Fridolin Simon Brand and Kurt Jax. Focusing the meaning(s) of resilience: Resilience as a descriptive concept and a boundary object. *Ecology and Society*, 12(1), 2007.

[69] Simon A. Levin and Jane Lubchenco. Resilience, robustness, and marine ecosystem-based management. *BioScience*, 58(1):27–32, January 2008.

[70] Jean M. Carlson and John Doyle. Complexity and robustness. In *Proceedings of the National Academy of Science*, volume 99, pages 2538–2545, 19 February 2002.

[71] Marie E. Csete and John C. Doyle. Reverse engineering of biological complexity. *Science*, 295:1664–1669, 1 March 2002.

[72] John D. Fernstrom and Richard J. Wurtman. Brain serotonin content: Physiological regulation by plasma neutral amino acids. *Science*, 178(4059):414–416, 27 October 1972.

[73] G. L. Stiles, M. G. Caron, and R. J. Lefkowitz. Beta-adrenergic receptors: Biochemical mechanisms of physiological regulation. *Physiological Reviews*, 64(2):661–743, April 1984.

[74] George F. Oster and Edward O. Wilson. *Caste and Ecology in the Social Insects.* Princeton University Press, 1978.

[75] Gene E. Robinson. Regulation of division of labor in insect societies. *Annual Review of Entomology*, 37:637–665, 1992.

[76] Deborah M. Gordon. The organization of work in social insect colonies. *Nature*, 380(6570):121–124, 14 March 1996.

[77] Charles Darwin. *The Origin of Species By Means of Natural Selection, or the Preservation of Favored Races in the Struggle for Life.* John Murray, 6th edition, 1876.

[78] David N. Reznick and Cameron K. Ghalambor. The population ecology of contemporary adaptations: What empirical studies reveal about the conditions that promote adaptive evolution. *Genetica*, 112–113(1):183–198, November 2001.

[79] Andreas Wagner. *Robustness and Evolvability in Living Systems.* Princeton University Press, 2005.

[80] Lawrence R. Walker and Roger del Morel. *Primary Succession and Ecosystem Rehabilitation.* Cambridge University Press, 2003.

[81] T. P. Hughes, A. H. Baird, D. R. Bellwood, M. Card, S. R. Connolly, C. Folke, R. Grosberg, O. Hoegh-Guldberg, J. B. C. Jackson, J. Kleypas, J. M. Lough, P. Marshall, M. Nyström, S. R. Palumbi, J. M. Pandolfi, B. Rosen, and J. Roughgarden. Climate change, human impacts and the resilience of coral reefs. *Science*, 301(5635):929–933, 15 August 2003.

[82] Mikhail L. Tsetlin. *Automaton Theory and the Modeling of Biological Systems.* Academic Press, 1973.

[83] Victor Varshavsky. *Collective Behavior of Automata.* Nauka, 1973. in Russian.

[84] K. S. Narendra and M. A. L. Thathachar. *Learning Automata: An Introduction.* Prentice Hall, 1989.

[85] Wen-Chiao Lin. *Rational Probabilistic Deciders.* PhD thesis, University of Michigan, 2006.

[86] Eric Bonabeau, Guy Theraulaz, Jean-Louis Deneubourg, Serge Aron, and Scott Camazine. Self-organization in social insects. *Trends in Ecology and Evolution*, 12(5):188–193, 5 May 1997.

[87] Simon A. Levin. Ecosystems and the biosphere as complex adaptive systems. *Ecosystems*, 1(5):431–436, September 1998.

[88] Edward L. Deci and Richard M. Ryan. *Intrinsic Motivation and Self-Determination in Human Behavior.* Plenum Press, 1985.

[89] Donald P. Green and Ian Shapiro. *Pathologies of Rational Choice Theory: A Critique of Applications in Political Science.* Yale University Press, 1994.

[90] Herbert A. Simon. *Models of Bounded Rationality*, volume 1–3. MIT Press, 1982, 1997.

[91] Ariel Rubinstein. *Modeling Bounded Rationality.* MIT Press, 1998.

[92] Andrew M. Colman. Cooperation, psychological game theory, and limitations of rationality in social interaction. *Behavioral and Brain Sciences*, 26(2):139–153, April 2003.

[93] Martin J. Osborne and Ariel Rubinstein. *A Course in Game Theory.* MIT Press, 1994.

[94] Gregor Mendel. Versuche über Plflanzen-hybriden (Experiments in plant hybridization). *Verhandlungen des naturforschenden Ver-eines in Brünn (Meetings of the Brünn Natural History Society)*, 4:3–47, 1865.

[95] Ronald A. Fisher. *The Genetical Theory of Natural Selection: A Complete Variorum Edition.* Oxford University Press, 1999.

[96] Sewall Wright. Evolution in Mendelian populations. *Genetics*, 160:97–159, 1931.

[97] J. B. S. Haldane. *The Causes of Evolution.* Princeton University Press, 1990.

[98] Warren J. Ewens. *Mathematical Population Genetics*, volume 1. Springer, 2nd edition, 2004.

[99] Carlos D. Bustamante. Population genetics of molecular evolution. In Rasmus Nielsen, editor, *Statistical Methods in Molecular Evolution*, Statistics for Biology and Health, pages 63–99. Springer, 2005.

[100] Jack Lester King and Thomas H. Jukes. Non-Darwinian evolution. *Science*, 164(3881):788–798, 16 May 1969.

[101] Motoo Kimura. *The Neutral Theory of Molecular Evolution.* Cambridge University Press, 1983.

[102] Manfred Eigen and Peter Schuster. The hypercycle: A principle of natural self-organization. *Naturwissenschaften*, 64(11):541–565, November 1977.

[103] Martin A. Nowak. What is a quasispecies? *Trends in Ecology and Evolution*, 7(4):118–121, April 1992.

[104] Mark A. Bedau, John S. McCaskill, Norman H. Packard, Steen Rasmussen, Chris Adami, David G. Green, Takashi Ikegami, Kunihiko Kaneko, and Thomas S. Ray. Open problems in artificial life. *Artificial Life*, 6(4):363–376, 2000.

[105] Scott Freeman. *Biological Science.* Pearson Prentice Hall, 2nd edition, 2005.

[106] David G. Green and David Newth. Towards a theory of everything? — Grand challenges in complexity and informatics. *Complexity International*, 8, January 2001.

[107] Bart Selman, Rodney A. Brooks, Thomas Dean, Eric Horvitz, Tom M. Mitchell, and Nils J. Nilsson. Challenge problems for artificial intelligence. In *Proceedings of AAAI-96, Thirteenth National Conference on Artificial Intelligence*, pages 1340–1345. AAAI Press, 4–8 August 1996.

[108] Balas Kausik Natarajan. *Machine Learning: A Theoretical Approach.* Morgan Kauffman, 1991.

[109] Tom M. Mitchell. *Machine Learning.* McGraw-Hill, 1997.

[110] King-Sun Fu. *Syntactic Methods in Pattern Recognition.* Academic Press, 1974.

[111] Pierre Kabamba. The von Neumann threshold of self-reproducing systems. In *Proceedings of the 2008 ASME Dynamic Systems and Control Conference*, 20–22 October 2008.

[112] Reinhard Diestel. *Graph Theory.* Springer, 3rd edition, 2005.

[113] Christos G. Cassandras and Stéphane Lafortune. *Introduction to Discrete Event Systems.* Springer, 2nd edition, 2008.

[114] Pierre Brémaud. *Markov Chains: Gibbs fields, Monte Carlo Simulation and Queues.* Springer, 1999.

[115] Daniel Burd. Plastic not fantastic. In *Project Reports of the Canada Wide Science Fair*, 2008.

[116] Günter Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, 5(1):96–101, 1994.

[117] David B. Fogel, Thomas Bäck, and Zbigniew Michalewicz. *Evolutionary Computation: Basic Algorithms and Operators*. CRC Press, 2000.

[118] Claude E. Shannon. A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423 and 623–656, July and October 1948.

[119] D. F. Kerridge. Inaccuracy and inference. *Journal of the Royal Statistical Society, Series B (Methodological)*, 23(1):184–194, 1961.

[120] Yunus A. Cengel and Michael A. Boles. *Thermodynamics: An Engineering Approach*. McGraw-Hill, 4th edition, 2001.

[121] R. K. Pathria. *Statistical Mechanics*. Butterworth-Heinemann, 2nd edition, 1996.

[122] E. T. Jaynes. Entropy and search theory. In *Proceedings of the First Maximum Entropy Workshop*, June 1981.

[123] E. T. Jaynes. Information theory and statistical mechanics. *The Physical Review*, 106(4):620–630, 15 May 1957.

[124] Dolph Schluter. Estimating the form of natural selection on a quantitative trait. *Evolution*, 42(5):849–861, September 1988.

[125] Joy Bergelson and Colin B. Purrington. Surveying patterns in the cost of resistance in plants. *The American Naturalist*, 148(3):536–558, September 1996.

[126] A. R. Kraaijveld and H. C. J. Godfray. Trade-off between parasitoid resistance and larval competitive ability in *Drosophila Melanogaster*. *Nature*, 389(6648):278–280, September 1997.

[127] Abel Bult and Carol B. Lynch. Breaking through artificial selection limits of an adaptive behavior in mice and the consequences for correlated responses. *Behavior Genetics*, 30(3):193–206, May 2000.

[128] Lingtian Xie and Paul L. Klerks. Responses to selection for cadmium resistance in the least killifish, *Heterandria Formosa*. *Environmental Toxicology and Chemistry*, 22(2):313–320, February 2003.

[129] Joep M. S. Burger, Munjong Kolss, Juliette Pont, and Tadeusz J. Kawecki. Learning ability and longevity: A symmetrical evolutionary trade-off in *Drosophila*. *Evolution*, 62(6):1294–1304, June 2008.

[130] Gene F. Franklin, J. David Powell, and Michael Workman. *Digital Control of Dynamic Systems*. Addison Wesley, 3rd edition, 1998.

[131] Arthur E. Bryson, Jr. and Yu-Chi Ho. *Applied Optimal Control: Optimization, Estimation, and Control*. Hemisphere, 1975.

[132] Andrew P. Sage and Chelsea C. White, III. *Optimum Systems Control*. Prentice Hall, 2nd edition, 1977.

[133] Robert F. Stengel. *Optimal Control and Estimation*. Dover, 1994.

[134] P. R. Kumar and Pravin Varaiya. *Stochastic Systems: Estimation, Identification, and Adaptive Control*. Prentice Hall, 1986.

[135] Karl J. Åström. *Introduction to Stochastic Control Theory*. Dover, 2006.

[136] P. Whittle. Multi-armed bandits and the Gittins index. *Journal of the Royal Statistical Society, Series B (Methodological)*, 42(2):143–149, 1980.

[137] Kenneth James Ryan and C. George Ray, editors. *Sherris Medical Microbiology: An Introduction to Infectious Diseases*. McGraw-Hill, 4th edition, 2004.

[138] Eldon John Gardner, Michael J. Simmons, and D. Peter Snustad. *Principles of Genetics*. Wiley, 8th edition, 1998.

[139] A. A. Barker. Monte Carlo calculations of the radial distribution functions for a proton-electron plasma. *Australian Journal of Physics*, 18:119–133, 1965.

[140] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[141] Karl Sims. Evolving 3d morphology and behavior by competition. In Rodney Brooks and Pattie Maes, editors, *Artificial Life IV: Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems*, pages 28–39. MIT Press, 1994.

[142] Madhav P. Desai and Vasant B. Rao. On the convergence of reversible Markov chains. *SIAM Journal on Matrix Analysis and Applications*, 14(4):950–966, October 1993.

[143] Pat Trizila, Chang-Kwon Kang, Miguel Visbal, and Wei Shyy. A surrogate model approach in 2D versus 3D flapping wing aerodynamic analysis. In *Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 10–12 September 2008.

[144] Peter Augustsson, Krister Wolff, and Peter Nordin. Creation of a learning, flying robot by means of evolution. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2002)*, 9–13 July 2002.

[145] Rusty Hunt, Gregory S. Hornby, and Jason D. Lohn. Toward evolved flight. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 2005)*, 25–29 June 2005.

[146] Yoon-Sik Shim and Chang-Hun Kim. Evolving physically simulated flying creatures for efficient cruising. *Artificial Life*, 12(4):561–591, 2006.

[147] John W. Roberts, Jun Zhang, and Russ Tedrake. Motor learning at intermediate Reynolds number: Experiments with policy gradient on a heaving plate. 2009.

[148] Nestor V. Queipo, Raphael T. Haftka, Wei Shyy, Tushar Goel, Rajkumar Vaidyanathan, and P. Kevin Tucker. Surrogate-based analysis and optimization. *Progress in Aerospace Sciences*, 41:1–28, 2005.

[149] J. Yan, R. J. Wood, S. Avadhanula, M. Sitti, and R.S. Fearing. Towards flapping wing control for a micromechanical flying insect. In *Proceedings of the 2001 IEEE International Conference on Robotics and Automation*, pages 3901–3908, 21–26 May 2001.

[150] Luca Schenato. *Analysis and Control of Flapping Flight: from Biological to Robotic Insects*. PhD thesis, University of California at Berkeley, 2003.

[151] Xinyan Deng, Luca Schenato, and S. Shankar Sastry. Flapping flight for biomimetic robotic insects: Part II -- flight control design. *IEEE Transactions on Robotics*, 22(4):789–803, August 2006.

[152] Jean-Baptiste Mouret, Stéphane Doncieux, and Jean-Arcady Meyer. Incremental evolution of target-following neuro-controllers for flapping-wing animats. In Stefano Nolfi, Gianluca Baldassarre, Raffaele Calabretta, John C. T. Hallam, Davide Marocco, Jean-Arcady Meyer, Orazio Miglino, and Domenico Parisi, editors, *From Animals to Animats 9*, volume 4095 of *Lecture Notes in Computer Science*, pages 606–618. Springer, 2006.

[153] Liguo Weng, Wenchuan Cai, M. J. Zhang, X. H. Liao, and David Y. Song. Neural-memory based control of micro air vehicles (mavs) with flapping wings. In Derong Liu, Shumin Fei, Zeng-Guang Hou, Huaguang Zhang, and Changyin Sun, editors, *Advances in Neural Networks — ISNN 2007*, volume 4491 of *Lecture Notes in Computer Science*, pages 70–80. Springer, 2007.

[154] Zaeem A. Khan and Sunil K. Agrawal. Control of longitudinal flight dynamics of a flapping-wing micro air vehicle using time-averaged model and differential flatness based controller. In *Proceedings of the 2007 American Control Conference*, pages 5284–5289, 11–13 July 2007.

[155] Soon-Jo Chung, Michael Dorothy, and Jeremiah R. Stonerz. Neurobiologically inspired control of engineered flapping flight. In *Proceedings of the 2009 AIAA Infotech at Aerospace and 2009 AIAA Unmanned Unlimited Conference*, number AIAA-2009-1929, 6–9 April 2009.

[156] Theodore Theodorsen. General theory of aerodynamic instability and the mechanism of flutter. Technical Report 496, NACA, 1935.

[157] I. E. Garrick. Propulsion of a flapping and oscillating airfoil. Technical Report 567, NACA, 1936.

[158] Harry R. Lewis and Christos H. Papadimitriou. *Elements of the Theory of Computation*. Prentice Hall, 2nd edition, 1998.

[159] E. Mark Gold. Language identification in the limit. *Information and Control*, 10(5):447–474, 1967.

[160] E. Mark Gold. Complexity of automaton identification from given data. *Information and Control*, 37(3):302–320, June 1978.

[161] Dana Angluin. On the complexity of minimum inference of regular sets. *Information and Control*, 39(3):337–350, December 1978.

[162] Leonard Pitt and Manfred K. Warmuth. The minimum consistent DFA problem cannot be approximated within any polynomial. *Journal of the Association for Computing Machinery*, 40(1):95–142, 1993.

[163] John E. Hopcraft, Rajeev Motwani, and Jeffrey D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison Wesley, 3rd edition, 2006.

[164] Michael J. Kearns and Umesh V. Vazirani. *An Introduction to Computational Learning Theory*. MIT Press, 1994.

[165] Rafael C. Gonzalez and Michael G. Thomason. *Syntactic Pattern Recognition: An Introduction*. Addison Wesley, 1978.

[166] Laurent Miclet. *Structural Methods in Pattern Recognition*. Chapman & Hall, 1986.

[167] Robin R. Murphy. *An Introduction to AI Robotics*. MIT Press, 2000.

[168] King-Sun Fu and Taylor L. Booth. Grammatical inference: Introduction and survey — Parts I and II. *IEEE Transactions on Systems, Man, and Cybernetics*, 5(1 and 4):59–72 and 409—423, January and July 1975.

[169] Laurent Miclet. Grammatical inference. In Horst Bunke and Alberto Sanfeliu, editors, *Syntactic and Structural Pattern Recognition: Theory and Applications*, volume 7 of *Series in Computer Science*, pages 237–290. World Scientific, 1990.

[170] Enrique Vidal. Grammatical inference: An introductory survey. In Rafael C. Carrasco and Jose Oncina, editors, *Grammatical Inference and Applications*, volume 862 of *Lecture Notes in Computer Science*, pages 1–4. Springer, 1994.

[171] Yasubumi Sakakibara. Recent advances of grammatical inference. *Theoretical Computer Science*, 185(1):15–45, October 1997.

[172] Colin de la Higuera. Current trends in grammatical inference. In Francesc J. Ferri, José M. Iñesta, Adnan Amin, and Pavel Pudil, editors, *Advances in Pattern Recognition: Joint IAPR International Workshops SSPR 2000 and SPR 2000*, volume 1876 of *Lecture Notes in Computer Science*, pages 28–31. Springer, 2000.

[173] Colin de la Higuera. A bibliographical study of grammatical inference. *Pattern Recognition*, 38(9):1332–1348, September 2005.

[174] B. A. Trakhtenbrot and Ya M. Barzdin. *Finite Automata: Behavior and Synthesis*. North-Holland, 1973.

[175] Pierre Dupont, Laurent Miclet, and Enrique Vidal. What is the search space of the regular inference? In Rafael C. Carrasco and Jose Oncina, editors, *Grammatical Inference and Applications*, volume 862 of *Lecture Notes in Computer Science*, pages 25–37. Springer, 1994.

[176] Kevin Lang. Random DFA's can be approximately learned from sparse uniform examples. In *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, pages 45–52. ACM Press, 27–29 July 1992.

[177] Kevin J. Lang, Barak A. Pearlmutter, and Rodney Price. Results of the Abbadingo One DFA learning competition and a new evidence-driven state merging algorithm. In Vasant Honavar and Giora Slutzki, editors, *Proceedings of the Fourth International Colloquium on Grammatical Inference (ICGI-98)*, volume 1433 of *Lecture Notes in Artificial Intelligence*, pages 1–12. Springer, 12–14 July 1998.

[178] Colin de la Higuera, Jose Oncina, and Enrique Vidal. Identification of DFA: Data-dependent versus data-independent algorithms. In Laurent Miclet and Colin de la Higuera, editors, *Grammatical Interference: Learning Syntax from Sentences*, volume 1147 of *Lecture Notes in Computer Science*, pages 313–325. Springer, 1996.

[179] Jean-Yves Giordano. Grammatical inference using tabu search. In Laurent Miclet and Colin de la Higuera, editors, *Grammatical Interference: Learning Syntax from Sentences*, volume 1147 of *Lecture Notes in Computer Science*, pages 292–300. Springer, 1996.

[180] Pierre Dupont. Regular grammatical inference from positive and negative samples by genetic search: The GIG method. In Rafael C. Carrasco and Jose Oncina, editors, *Grammatical Inference and Applications*, volume 862 of *Lecture Notes in Computer Science*, pages 236–245. Springer, 1994.

[181] Jon Louis Bentley and Andrew Chi-Chih Yao. An almost optimal algorithm for unbounded searching. *Information Processing Letters*, 5(3):82–87, August 1976.

[182] J. C. Raoult and J. Vuillemin. Optimal unbounded search strategies. In J.W. de Bakker and J. van Leeuwen, editors, *Automata, Languages and Programming*, volume 85 of *Lecture Notes in Computer Science*, pages 512–530. Springer, 1980.

[183] Richard Beigel. Unbounded searching algorithms. *SIAM Journal on Computing*, 19(3):532–537, June 1990.

[184] Hassan K. Khalil. *Nonlinear Systems.* Prentice Hall, 3rd edition, 2002.

[185] Madan M. Gupta, Liang Jin, and Noriyasu Homma. *Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory.* Wiley, 2003.