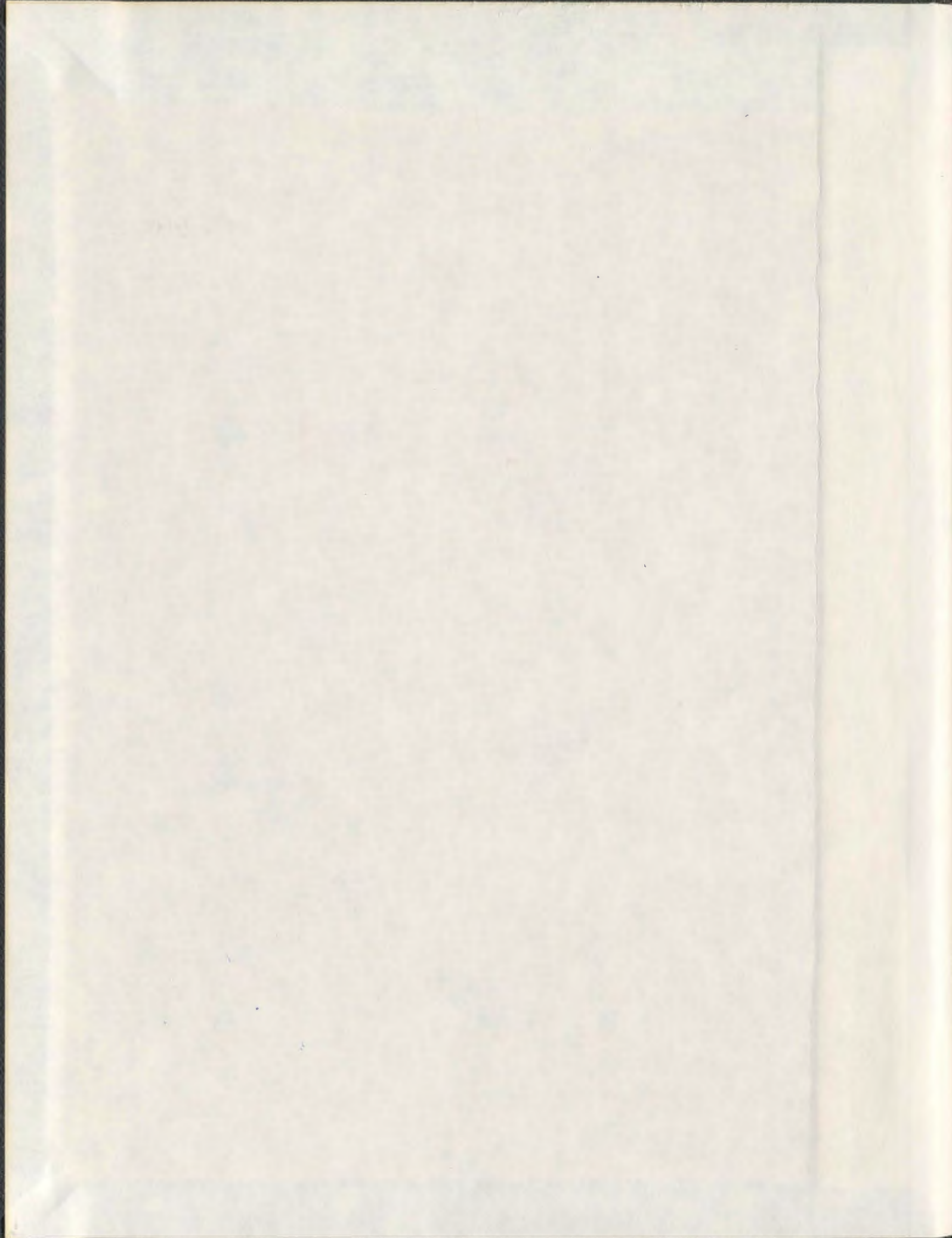


EVOLVABILITY AND RATE OF EVOLUTION
IN EVOLUTIONARY COMPUTATION

TING HU



001311



Evolvability and Rate of Evolution in Evolutionary Computation

by

© Ting Hu

A thesis submitted to the
School of Graduate Studies
in partial fulfilment of the
requirements for the degree of
Doctor of Philosophy

Department of Computer Science
Memorial University

May 2010

St. John's

Newfoundland

Abstract

Evolvability has emerged as a research topic in both natural and computational evolution. It is a notion put forward to investigate the fundamental mechanisms that enable a system to evolve. A number of hypotheses have been proposed in modern biological research based on the examination of various mechanisms in the biosphere for their contribution to evolvability. Therefore, it is intriguing to try to transfer new discoveries from Biology to and test them in Evolutionary Computation (EC) systems, so that computational models would be improved and a better understanding of general evolutionary mechanisms is achieved.

Rate of evolution comes in different flavors in natural and computational evolution. Specifically, we distinguish the rate of fitness progression from that of genetic substitutions. The former is a common concept in EC since the ability to explicitly quantify the fitness of an evolutionary individual is one of the most important differences between computational systems and natural systems. Within the biological research community, the definition of rate of evolution varies, depending on the objects being examined such as gene sequences, proteins, tissues, etc. For instance, molecular biologists tend to use the rate of genetic substitutions to quantify how fast evolution proceeds at the genetic level. This concept of rate of evolution focuses on the evolutionary dynamics underlying fitness development, due to the inability to mathematically define fitness in a natural system. In EC, the rate of genetic substitutions suggests an unconventional and potentially powerful method to measure the rate of evolution by accessing lower levels of evolutionary dynamics.

Central to this thesis is our new definition of rate of evolution in EC. We trans-

fer the method of measurement of the rate of genetic substitutions from molecular biology to EC. The implementation in a Genetic Programming (GP) system shows that such measurements can indeed be performed and reflect well how evolution proceeds. Below the level of fitness development it provides observables at the genetic level of a GP population during evolution. We apply this measurement method to investigate the effects of four major configuration parameters in EC, i.e., mutation rate, crossover rate, tournament selection size, and population size, and show that some insights can be gained into the effectiveness of these parameters with respect to evolution acceleration. Further, we observe that population size plays an important role in determining the rate of evolution. We formulate a new indicator based on this rate of evolution measurement to adjust population size dynamically during evolution. Such a strategy can stabilize the rate of genetic substitutions and effectively improve the performance of a GP system over fixed-size populations. This rate of evolution measure also provides an avenue to study evolvability, since it captures how the two sides of evolvability, i.e., variability and neutrality, interact and cooperate with each other during evolution. We show that evolvability can be better understood in the light of this interplay and how this can be used to generate adaptive phenotypic variation via harnessing random genetic variation. The rate of evolution measure and the adaptive population size scheme are further transferred to a Genetic Algorithm (GA) to solve a real world application problem - the wireless network planning problem. Computer simulation of such an application proves that the adaptive population size scheme is able to improve a GA's performance against conventional fixed population size algorithms.

Acknowledgements

First I wish to thank my parents for bringing me into this world and encouraging me for all the years. Millions of thanks to them for their constant support.

I am very grateful to my doctoral supervisor, Dr. Wolfgang Banzhaf, who offered me the chance joining his group and showing me how a genuine scientist thinks and speaks in multiple scientific languages. His great insights of science inspire me to start this research. Without his supervision and constant support in the past four and a half years, I would not be able to bring my research to this successful end. He has being a great mentor not only for my Ph.D, but also for my whole life.

There are other persons in the Department of Computer Science, Memorial University that I wish to acknowledge here. Thank you Dr. John Shieh and Dr. Sharene Bungay, my Ph.D. program committee, for the help on my research and writing the thesis. Thank you Dr. Yuanzhu Peter Chen, for all the advices on academic writing and how to become a good researcher. Thanks to Dr. Simon Harding and Dr. Garnett Wilson for the advices on my research. Thank you Ms. Elaine Boone and Ms. Darlene Oliver for taking good care of everything. I really appreciate all the support and resources from the department and the university.

I also would like to thank my thesis examiners, Dr. Steve Carr, Dr. Minglun Gong, and Dr. Christian Jacob. Their invaluable suggestions are very appreciated.

Finally I wish to thank all my friends and peers in St. John's and other places all over the world. You brought many wonderful memories and much fun to my life.

Contents

Abstract	ii
Acknowledgements	iv
List of Tables	ix
List of Figures	x
1 Introduction	1
1.1 Contribution	2
1.2 Overview	6
2 Evolvability	8
2.1 Understanding evolvability	8
2.2 Complexity and evolvability in the biosphere	11
2.2.1 Representation	11
2.2.2 Exploratory operations	19
2.2.2.1 Genotypic variation	20
2.2.2.2 Phenotypic variation	26

2.2.2.3	Epigenetic mechanism	32
2.2.3	Stabilization operations	34
2.2.3.1	Recombination	34
2.2.3.2	Selection	37
2.2.4	Quality differentiation	39
2.2.4.1	Genotype-phenotype mapping	39
2.2.4.2	Fitness evaluation	46
2.3	Discussion	48
3	Rate of Evolution	50
3.1	Measuring rate of evolution	51
3.2	The k_a/k_s ratio in Biology	53
3.3	Applying k_a/k_s ratio in EC	57
3.3.1	Case study: tree-based Genetic Programming	59
3.3.2	Simulation with symbolic regression	62
3.3.2.1	Single evolution process	63
3.3.2.2	Comparisons with varying parameter setting	66
3.4	Discussion	72
4	Role of Population Size in Rate of Evolution	73
4.1	Background and motivation	74
4.1.1	Population size control	75
4.1.2	Population size and rate of evolution	76
4.2	Adjusting population size during evolution	78
4.2.1	Adjustment indicator: rate of evolution	80

4.2.2	Adaptive population size approach	81
4.3	Simulation with tree-based GP	82
4.3.1	Test suite: Mackey-Glass time series	82
4.3.2	Results	84
4.4	Discussion	89
5	Neutrality and Variability: Two Sides of Evolvability	90
5.1	Evolvability and environment	91
5.2	Methods	93
5.2.1	The k_a/k_s ratio for Linear GP	94
5.2.2	Neutral networks	94
5.3	Simulated studies with Linear GP	96
5.3.1	Test case: polynomial regression	96
5.3.2	Varying environmental scenarios	97
5.3.2.1	Random evolution (RE)	98
5.3.2.2	Fixed target (FT) evolution	100
5.3.2.3	Moving target (MT) evolution	105
5.4	Discussion	109
6	Real-World Application	111
6.1	Wireless network planning problem	112
6.1.1	Problem description	113
6.1.2	Problem formulation	113
6.2	Evolutionary approach to network planning	117
6.2.1	Individual representation	118

6.2.2	Evolution framework	119
6.2.3	Adaptive population size	120
6.2.4	Evolutionary operations	122
6.3	Simulation	124
6.3.1	Network layouts	125
6.3.2	Results	126
6.4	Discussion	129
7	Concluding Remarks	131
7.1	Summary	131
7.2	Future research	133
	Bibliography	137

List of Tables

4.1	Tree-Based GP configuration for the Mackey-Glass time series.	84
4.2	Best fitness ($\times 10^{-3}$) comparison with different sized populations. . .	87
5.1	Liner GP configuration for polynomial regression.	97
6.1	Network configurations.	125
6.2	Results of APS-GA (average data over 100 runs).	128
6.3	Results of FPS-GA (average data over 100 runs).	129

List of Figures

2.1	The genetic switch in Jacob-Monod model (adapted from [89]).	17
2.2	The Central Dogma of Molecular Biology.	40
2.3	Eukaryotic genetic system.	44
3.1	Codon-based sequence comparison.	55
3.2	The flow chart with measuring rate of evolution in GP.	59
3.3	Fitness and diversity development in a single GP evolution.	64
3.4	The k_a/k_s measurement in a GP evolution.	64
3.5	Evolution with different tournament selection sizes.	68
3.6	Evolution with different population sizes.	69
3.7	Evolution with different mutation rates.	70
3.8	Evolution with different crossover rates.	71
4.1	Positive selection in different sized populations.	79
4.2	Negative selection in different sized populations.	79
4.3	Mackey-Glass time series.	83
4.4	An example GP run with adaptive population size.	85
4.5	Correlations between population size and k_a/k_s ratio.	86

4.6	Solution quality and computation costs comparisons.	88
5.1	An example of neutral networks.	95
5.2	A typical single run of random evolution.	99
5.3	A typical single run of fixed target evolution on $x^2 + x$	101
5.4	A typical single run of fixed target evolution on $x^4 + x^3 + x^2 + x$ for the first 200 generations.	103
5.5	Variability measurements in random evolution and fixed target evolution.	104
5.6	A typical single run of moving target evolution.	106
5.7	Three typical cases of moving target evolution.	107
6.1	Example of a wireless network layout.	114
6.2	Example of a network planning for the network layout in Figure 6.1. .	114
6.3	Two-tier chromosome representation.	118
6.4	Four network layouts in two different scenarios.	126
6.5	Examples of best solutions to the four network layouts.	127

Chapter 1

Introduction

Inspired by the evolution process in nature, the field of *Evolutionary Computation* (EC) has seen substantial progress since it was founded in the 1960's and 1970's. [16, 48, 49, 58, 73, 91, 143, 156].

In EC, candidate solutions to optimization or learning problems are represented by structures similar to gene sequences of Biology and their phenotypic expressions. The ensemble of such solutions is referred to as a population. Evolutionary operators, such as mutation, recombination, and selection are applied to this population. Solutions gradually improve by repeating a variation-selection cycle in the evolutionary process through numerous iterations. Essentially a search method, EC often produces well-performing solutions to complex optimization and learning problems and is applied in various areas.

The fundamental idea of EC was gleaned from Biology, and more specifically, from Darwin's theory of evolution by natural selection [34] as embodied in the neo-darwinian synthesis [63, 116]. However, knowledge of natural evolution has deepened

profoundly in Biology in the past decades. This progress has, to a large degree, not been incorporated yet into computational models of evolution, and therefore cannot be harvested in applications. It has been realized in the literature that adopting new developments from areas such as molecular genetics, cell biology, developmental biology and evolutionary biology would substantially benefit EC [13, 169].

The question then arises what the most important and revolutionary discoveries in Biology are in recent times, and how they could be sufficiently abstracted to provide material for computational models. As the number of scientists working in the areas mentioned above is now higher than at any other time in history, it becomes non-trivial to select those aspects of evolution that will have the most impact in computational models. A number of books have appeared in recent years that provide some guidance in this quest (see, e.g., [27, 51, 80, 89, 135, 174]), and here we are mainly interested in the concepts of evolvability and the rate of evolution.

1.1 Contribution

The primary goal of this thesis is to transfer some discoveries with regard to evolvability and rate of evolution from the modern biological literature to the area of EC. This thesis contributes to the research of EC in the following ways.

- It discusses the concept of evolvability with the background suitable for its understanding in both natural and computational evolution.
- It reviews new developments in Biology regarding the concept of evolvability. It is expected that these new aspects are worth incorporation in computational

models through additional sophisticated mechanisms useful for tackling complex application problems.

- It introduces a measure for rate of evolution from molecular biology that is able to capture the evolution dynamics at the genetic level underneath fitness progression. This measurement method is then transferred to EC to allow the capability to capture the rate of genetic substitutions, providing a different means to observe evolution.
- It investigates the effectiveness of configuration parameters in an EC algorithm with regard to rate of evolution. The mutation rate, crossover rate, and tournament selection size are shown to have a monotonic relationship with the rate of genetic substitution, while population size presents a more complex relationship.
- It further examines the role of population size in the rate of evolution. Inspired by the findings in modern population genetics that population size is crucial in relation to the exploration of neutrality in an evolutionary process, an adaptive population size adjustment method is proposed, again based on the rate of evolution measurement. This method is shown to effectively improve EC performance compared to fixed-size populations in the context of a GP problem and further on a real world application with GA .
- Motivated by biological discoveries and analysis, it proposes an understanding of evolvability based on how its two sides, neutrality and variability, closely cooperate with each other to facilitate the use of random genetic variations for adaptive phenotypic variations. The proposed measurement of rate of evolution

turns out to be a powerful tool to study evolvability from this perspective.

Partial contents of this thesis and related research have been published in journals and conference proceedings as follows:

- **Ting Hu** and Wolfgang Banzhaf. Evolvability and Speed of Evolutionary Algorithms in the Light of Recent Developments in Biology. *Journal of Artificial Evolution and Applications*, in press.
- **Ting Hu**, Simon Harding, and Wolfgang Banzhaf. Variable Population Size and Evolution Acceleration: A Case Study with A Parallel Evolutionary Algorithm. *Genetic Programming and Evolvable Machines*, 11(2):205-225, 2010.
- **Ting Hu**, Yuanzhu Peter Chen, and Wolfgang Banzhaf. WiMAX Network Planning Using Adaptive-Population-Size Genetic Algorithm. *The 7th European Event on the Application of Nature-inspired Techniques for Telecommunication Networks and other Parallel and Distributed Systems (EvoCOMNET in EvoStar 10)*. LNCS, vol. 6025, pp. 31-40.
- **Ting Hu** and Wolfgang Banzhaf. Neutrality and Variability: Two Sides of Evolvability in Linear Genetic Programming. *The 11th Genetic and Evolutionary Computation Conference (GECCO 09)*, pp. 963-980.
- **Ting Hu**, Yuanzhu Peter Chen, Wolfgang Banzhaf and Robert Benkoczi. An Evolutionary Approach to Planning IEEE 802.16 Networks. *The 11th Genetic and Evolutionary Computation Conference (GECCO 09)*, pp. 1929-1930.

- **Ting Hu** and Wolfgang Banzhaf. The Role of Population Size in Rate of Evolution in Genetic Programming. *The 12th European Conference on Genetic Programming (EuroGP 09)*, LNCS, vol. 5481, pp. 85-96.
- **Ting Hu** and Wolfgang Banzhaf. Nonsynonymous to Synonymous Substitution Ratio k_a/k_s : Measurement for Rate of Evolution in Evolutionary Computation. *The 10th International Conference on Parallel Problem Solving from Nature (PPSN X)*, LNCS, vol. 5199, pp. 448-457.
- **Ting Hu** and Wolfgang Banzhaf. Measuring Rate of Evolution in Genetic Programming Using Amino Acid to Synonymous Substitution Ratio k_a/k_s . *The 10th Genetic and Evolutionary Computation Conference (GECCO 08)*, pp. 1337-1338.

1.2 Overview

Chapter 2 discusses the concept of evolvability from the viewpoints of Biology and EC. A review on new developments in evolvability in the biosphere is presented. Its goal is to provide a background of general principles in biological evolution which inspires more complex and intelligent mechanisms in computational models. Biological notions are introduced to the degree that they correspond to their counterparts in EC.

Chapter 3 focuses on the rate of evolution. It starts by discussing different approaches to measuring the rate of evolution in both Biology and EC. A very important rate of evolution measurement, i.e., the k_a/k_s ratio, is introduced and transferred from molecular biology. This measurement method is tested on a tree-based GP system, and four major configuration parameters: mutation rate, crossover rate, tournament selection size, and population size, are examined with regard to their effectiveness in changing the rate of evolution.

Unlike the other three parameters, population size has a unique relationship to the rate of evolution. This relationship is studied in depth in Chapter 4. Following theoretical studies in population genetics, an adaptive population size approach is proposed to adjust the population size dynamically during evolution. The rate of evolution measurement provides the feedback signal in this mechanism. Simulations verify that adjusting the population size according to the rate of evolution is an effective approach to accelerate evolution. The exploration of neutrality under varying selection pressure is proposed to play a central role in this effect.

The core of evolvability, namely the ability to generate adaptive phenotypic varia-

tion from random genetic variation, is produced by the close cooperation of neutrality and variability. In Chapter 5, the measurement of rate of evolution is applied to study evolvability. It is shown how the two sides of evolvability, neutrality and variability, interact with each other over time in a single evolutionary process to keep the system both tolerant to deleterious genetic changes and sensitive to useful phenotypic adaptation.

A wireless network planning problem is employed as a real world application of Genetic Algorithm (GA) with our adaptive population size scheme in Chapter 6. It is known that GA can have variants when applied to various optimization problems. For the particular wireless network planning problem, novel individual representation and evolutionary operations are carefully designed to embrace the properties of such an application. In addition, the rate of evolution measure k_a/k_s ratio and its supervised population size adjustment are applied to the GA in this context. This aims to explore the central idea of this thesis on real world application, and further demonstrates the effectiveness of our methods to improve the performance of EC systems.

Concluding remarks and future research directions of our work are discussed in the last chapter.

Chapter 2

Evolvability

In the process of evolution, genetic variation explores new evolutionary material, the corresponding phenotypic variation provides adaptive characteristics, and stabilization operators like recombination and selection preserve these improvements over the previous generations. It is the interactions among these operations that allow evolution to work. Thus, the evolvability of an evolutionary system is constituted by the capability to coordinate these operations in such a way that phenotypic changes occur. A growing number of efforts have been dedicated to understanding [103, 141] and enhancing [35, 177] evolvability.

2.1 Understanding evolvability

Evolvability is the potential of a population to evolve. While the concept of evolvability is still very much under discussion, we want to venture to propose a definition that is equally applicable to natural and artificial systems:

Definition 1: Evolvability. The capacity of a population to generate adaptive phenotypic variation under certain environmental conditions and to transmit it to the next generation via an evolutionary process.

Altenberg [5] describes evolvability from a viewpoint of EC as the ability of a genetic operator or representation scheme to produce offspring fitter than their parents. In Biology, Kirschner and Gerhart [88] suggest that evolvability should be understood as an organism's capacity to generate heritable and selectable phenotypic variation. An explicit comparison between evolvability of biological and computational systems has been performed by Wagner and Altenberg [177]. In their view, evolvability should be considered as the ability of random variants to produce occasional improvements, which depends critically on the plasticity of the genotype-phenotype mapping. Marrow [110] suggests that evolvability means the capability to evolve, and this characteristic should be relevant to both natural and artificial evolutionary systems.

Recently, a growing number of evolutionary biologists and computer scientists have shown interests in this topic. In an evolutionary system, many properties of a population are considered related to evolvability, including adaptive representation [145], facilitation of extra-dimensional bypass and robustness against genetic variability [31, 175], redundancy and flexibility during developmental processes [88], mutation rate adaptation [21].

The detection and investigation of evolvability are non-trivial and intriguing problems. Phenotypic fitness is directly observable and serves as a selection criterion. However, as a *potential* to generate better fitness and a *capability* for adaptive evolu-

tion, evolvability is difficult to observe and to quantify. Although a formal method to measure evolvability has not yet been agreed upon in the literature, some empirical methods have been proposed. Nehaniv [121] suggests using evolutionary system complexity to describe and measure evolvability. He specifies the *exhibited evolvability* as an observable outcome generated by evolvability, and measures evolvability by the rate of the increasing complexity of evolutionary entities in an evolutionary system. Wagner [176] proposes to measure simply the amount of non-neutral 1-step mutation variation in a biological system (RNA) of particular relevance to evolution in order to quantify evolvability.

Earl and Deem [39] suggest that evolvability can be selected for by varying the environment. By observing genetic changes in protein evolution, they find that rapid or dramatic environmental change generates strong selection pressure for evolvability. Thus, high evolvability can be detected and favored by such selection pressure. Reisinger et al. [145, 146] propose an indirect encoding representation to improve the evolvability by its capacity to facilitate effective search. A gradually changing fitness function is designed to measure evolvability of such a representation and to evolve a population that is adaptive under different environments. Furthermore, as the pace of change of the fitness function increases, stronger selection pressure for evolvability is imposed.

From the above introduction, it can be seen that the research on evolvability is still very much under discussion. This motivates us to first visit biosphere to study how natural systems possess incredible complexity, and why they can have much higher evolvability than simulated evolutionary systems in EC.

2.2 Complexity and evolvability in the biosphere

In this section, we discuss evolvability by drawing ideas from complex natural systems. Notions and new discoveries from Biology are introduced, including aspects that have been recognized to improve evolvability in natural systems. The order of the presentation of these ideas conforms to the flow of an evolutionary algorithm. A brief discussion of their potential in designing new models in EC will also be given. We start with the characteristics of individual representation. Exploratory and stabilization operations are then investigated respectively, followed by quality differentiation.

2.2.1 Representation

The first step for setting up evolution with a population is to decide on the representation of evolutionary individuals. Each individual should be encoded as a candidate solution to a given problem, which subsequently determines the search space of an algorithm. Therefore, the representation strategy is important because it predicates the input to the search process that should produce a satisfactory output. Here, we highlight two biological mechanisms, a protection mechanism for robust information preservation, and a communication mechanism for information interaction between different molecules. First we review general forms of redundancy in living systems. Then we discuss molecular interaction to encourage communication among different components of an individual.

Redundancy

Living systems may seem wasteful and luxurious to computer scientists. The most distinguishing aspects of Biology compared to other natural sciences are complexity and diversity, which are indeed of central concern to biologists. In the face of competitive circumstances, organisms show great redundancy and resilience. Redundancy exists at different levels in natural organisms, including the genomic, transcriptomic, and phenotypic levels.

In Biology, the genome of an organism is defined as the information encoded in DNA sequences and inherited from generation to generation. The double-helical structure of DNA sequences itself is a form of protective redundancy of genetic information. Genomes carry genes and other non-coding DNA sequences. A gene is a string of base pairs grouped by a function that is embodied in a protein or polypeptide (protein fragment). Non-coding DNA sequences, formerly called "junk DNA", are not expressed as proteins, although they might be transcribed into RNA and involved in manufacturing proteins or controlling that process. However, genes are only quite small a fraction of the entire genome [148], with more than 98% of the human genome, for instance, being non-coding DNA sequences [9]. Furthermore, even a gene sequence itself is divided into exons and introns, where exons directly determine the protein amino acid sequence but introns do not. Nevertheless, these non-coding DNA sequences are not useless. Recent biological discoveries show that they play an important role in the regulation of gene transcription [185]. Regulation mechanisms will be discussed later in Section 2.2.4.1. Wren et al. [186] find that tandem-repeat polymorphisms in genes are quite common, and that such polymorphisms can enhance

the ability of some genes to respond rapidly to fluctuating selection pressure. The mechanism of gene duplication will be discussed in detail in Section 2.2.2.1. Moreover, diploid organisms have two copies of each chromosome, one copy inherited from each parent. Recent research has also found that a great number of DNA segments appear in more than two copies. *Copy Number Variations* (CNVs) in human and other mammalian genomes discovered lately accounts for a substantial amount of genetic variation other than single nucleotide polymorphisms [53, 79, 157, 171]. They are considered to have an important contribution to phenotypic variation, a phenomenon that will be discussed in detail later in Section 2.2.2.1.

The transcriptome describes the set of all transcribed RNAs in a particular cell. In the human transcriptome, the portion of transcribed non-protein-coding sequences is large and shows great complexity [54]. Substantially more DNA is transcribed than is translated, and only a small proportion of hnRNAs is translated into proteins. The rest is called *non-coding RNA* or ncRNA. About 98% of all transcribed sequence in humans is of this type [112]. Although many of the functions of these non-coding sequences are unclear, the high complexity of the transcriptome hints at its importance in the mechanisms of gene transcription.

As another important contributor to evolvability, redundancy at the level of the genome and of the transcriptome has attracted increasing research interest in evolutionary biology [36], and was found to be created by a number of mechanisms. Krakauer and Plotkin [94] go further and propose the new concept of *antiredundancy*. In their opinion antiredundancy emerges as does redundancy in cells, and natural organisms would be able to modify the redundancy properties of genotypes during evolution. Mechanisms for redundancy mask the phenotypic effect of mutations and

allow mutants to stay in populations, while mechanisms for antiredundancy enhance the efficiency of local selection to remove damaged components.

Redundancy at the phenotypic level lies in an organism's robustness against intrinsic or environmental changes. We adopt Wagner's definition here:

Definition 2: Robustness. The robustness of a biological or engineering system is its capability to continue functioning in the face of genetic or environmental perturbations [174].

It seems that robustness and evolvability have a contradictory relationship to each other. When a system has high robustness in its genome, it can be tolerant to intrinsic or environmental changes, but that should leave it less evolvable, as variation would be masked, and vice versa. In recent contributions, Wagner [175, 176] resolves this apparent contradiction. He distinguishes robustness and evolvability as quantities at both the genotypic and the phenotypic level. If one considers genotype, the more robust a genetic sequence is, the less innovation this sequence will produce. However, robustness and evolvability are characteristics of an entire system and if investigated at the phenotypic level show a strong correlation. A system with high phenotypic robustness harbors a great number of "neutral" variations that have no functional effects. These neutral variations do not change phenotypic function during relatively static evolutionary periods but may be able to generate adaptation later under certain genetic or environmental changes. Thus, a system with high phenotypic robustness simply masks changes but provides great potential for phenotypic innovation in the future, e.g. if conditions change and previously neutral changes suddenly have an effect. This is the core of the argument that high robustness and high evolvability

are in fact correlated in nature [176].

Redundancy is wide-spread in natural organisms as an efficient protection mechanism against internal or environmental changes, whereas in EC models components that do not seem to be immediately relevant are often considered superfluous. In recent years, however, representation redundancy has arisen as a by-product of computational evolution and has attracted increasing interest from EC researchers.

Definition 3: Representation Redundancy. In genetic and evolutionary algorithms, representations are redundant if the number of genotypes exceeds the number of phenotypes [151].

Rothlauf and Goldberg [151] examine the effects of redundant representations on the performance of an EC system both theoretically and empirically, and propose that redundant representations can increase the reliability and efficiency of EC models. Specifically in genetic programming, representation redundancy is usually identified as *introns* (formerly recognized as non-effective, neutral code) [16] in programs. Researchers have investigated both the positive and negative effects of introns [100, 107, 128, 189], and a positive relation between neutral code and evolvability in genetic programming has been suggested. The important role of redundancy in evolvability has now been realized. We might, therefore, consider designing protective redundancy into our algorithms to make them resilient against changes while improving adaptivity. Such capabilities certainly complicate the algorithms but may be worthwhile if the resulting robustness can generate higher evolvability when applying intense pressure to produce adaptive responses. Evolution might even be accelerated because the system has a quick and robust “reply” to evolutionary pressures. With

the growth of computational power available today ideas like these can be more easily explored than before.

Molecular interaction

Natural living systems are remarkably diverse ranging from so simple organisms as bacteria to highly complex creatures such as eukaryotes. This diversity is not the result of vastly different chemical constituents of organisms. In fact, many species carry out similar metabolic, cell division and replication processes under similar assembly principles [18]. The differences that distinguish species are caused by the regulation of basic coding genetic sequences [81] and molecular interactions contribute significantly to these organizational mechanisms.

Molecular interactions in a cell happen between the same type of molecules, such as protein-protein interactions, or between different types of molecules, such as protein-DNA or RNA-protein interactions. Signals can also be sent between and responded to by cells in multicellular organisms. Molecular interactions can be triggered by energy supply, e.g., in *metabolic pathways*, chains of interactions catalyzed by enzymes, or triggered by external stimuli, e.g., *signaling pathways* that enable communication through the cell membrane [29]. Proteins are not only a product enabling various organismal structures, but also work as control factors in various processes from the synthesis of a cell, to metabolism, gene regulation, and sexual reproduction.

Metabolism is a key process to maintain the growth and reproduction of cells. The metabolic network of a cell is an elaborate set of numerous chemical reactions catalyzed by enzymes [178]. Different types and amounts of enzymes are produced according to different energy supplies, and these enzymes will determine different

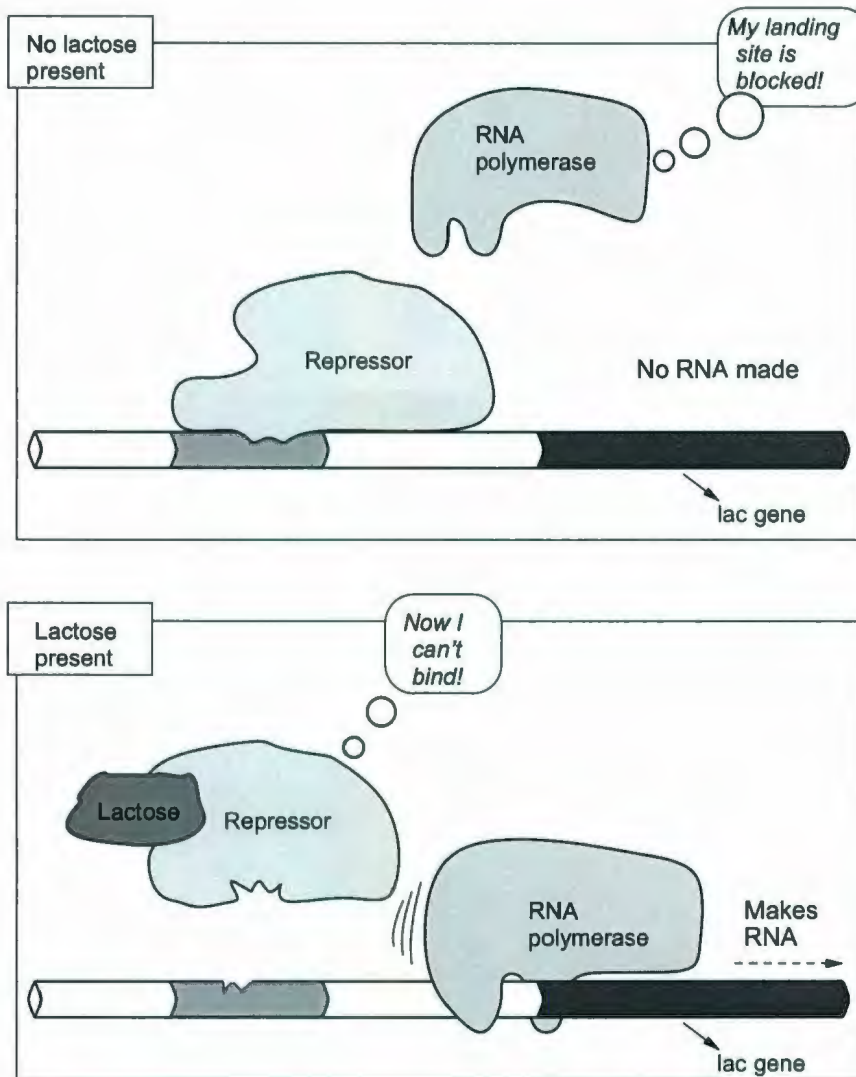


Figure 2.1: The genetic switch in Jacob-Monod model (adapted from [89]).

metabolic pathways by their catalysis. In the process of gene expression, the function achieved can be controlled by molecular interactions. For instance, the process of how a parsimonious bacterium responds to food supplies during metabolism shows a simple genetic switch mediated by molecular interactions. Since the metabolic pathways of bacteria are much simpler than those of multicellular organisms, the regulation of gene expression is more easily understandable in bacteria. The phenomenon of

enzyme induction [89] describes the adaptation of a bacterium to material supplies by producing varying amounts of enzyme. What triggers this production and how does this mechanism work? The Jacob-Monod model (shown in Figure 2.1) first described the regulation mechanism of inhibiting or repressing genes by inhibitory proteins, called *repressors* in bacteria. The binding of lactose to a repressor enables the production of RNAs by removing the repressor from its binding sites on the gene sequence where RNA polymerase can bind. However, this is not a simple on-off switch model. The continuity lies in the binding duration which determines the rate of protein synthesis. Therefore, if more sugar is absorbed during metabolism, more protein is synthesized by RNA translation. This simple sugar metabolism model captures the mechanism of how a repressor affects gene function. The enzyme here works as a trigger for the protein synthesis process under various molecular interactions. In addition, most enzyme effects are sensitive to ambient temperature [1], which is an important parameter to control metabolic interactions.

Signaling and cellular responses to signals are complex. These responses are controlled by a plethora of positive and negative feedback loops. The presence of feedback complicates the simple picture of a linear pathway, but is an essential part of the signaling process [18]. This makes signaling pathways involving molecular or cellular communication a network-like structure, with complex regulatory processes at work. The linkage between various parts of the gene expression apparatus in eukaryotic organisms is weakened by a far less-precisely defined control than that found in prokaryotic cells [88]. For instance, geometric requirements for binding sites (the exact binding locations) are significantly relaxed in eukaryotic gene regulation. A repressor does not have to bind at the exact position of a target, but need only bind

in the neighborhood. By lowering constraints for cooperation, such a weak linkage also enables potential interactions between different gene sequences. Signaling between cells is possible only after a sufficiently large number of repressors participate simultaneously. A single signal may incur a very complex response [31]. Allosteric proteins, which have multiple sites for interaction, also make gene expression more flexible because they have different sites for different functions. Regulatory decisions on which genes are transcribed when, where, and under what circumstances makes eukaryotic cells well conserved but enormously adaptive to generate new phenotypes in changing environments [106].

Computational models have already been used to analyze and understand complex multi-input/output and higher-order signaling systems have been examined in bioinformatics [126]. In contrast, current EC models are mostly limited to representing evolutionary material based on the infrastructure of natural organisms, while disregarding the vast potential of interaction mechanisms for regulation and signaling at both the molecular and cellular levels. The absence of such mechanisms in EC, however, points to significant research opportunities in this area.

2.2.2 Exploratory operations

Evolvability is understood as the capability to generate offspring fitter than their parents [5]. Exploratory operators are a main aspect of that capability, since they generate the necessary variation among individuals. Due to the complex mapping process from genotypic to morphological level in Biology, genotypic and phenotypic variation are discussed separately.

2.2.2.1 Genotypic variation

Genotypic variation is the result of changes of DNA sequences in both protein-coding and non-coding regions in the form of point mutation and gene rearrangements. Gene sequences are highly conserved against lethal changes that would likely lead to destructive consequences, otherwise a tiny mutant at the genetic level can cause a great change in function [89]. In contrast, changes to the regulatory or non-coding part of sequences are considered more likely to increase adaptability and plasticity of a system. In this section, we will discuss the general form of mutation first, then gene duplication as a very most important form of genetic variation followed by a comparison between point mutation and gene rearrangement.

Mutation

Since Darwin declared that natural selection explains evolution, controversies have arisen on different aspects of this explanation. In modern Biology, the two main schools of thought were selectionism and neutralism [124]. Some scientists argue that genotypic variation is maintained by selection, which is the central perspective of *neo-Darwinians*. Other evolutionists insist that high genotypic variation can be explained as a result of neutral mutations. In either case, mutation is accepted as one of the major mechanism to generate genotypic variation.

Mutation can happen at either coding or non-coding regions of DNA sequences, and may consequently cause functional or regulatory changes. The notion of neutral mutation is based on the fact that the majority of mutations have no consequent effects on protein function due to the redundancy in the translation apparatus, i.e.,

degeneracy of code. By observing the rate of nucleotide substitution, neutralism proposes that mutations change the function of gene products barely noticeably [124].

What triggers mutation and what is the relation between mutation and selection? Does selection pressure indeed generate new mutations or simply allow existing mutants to be fixed faster than before? Research on mutation under selection has received wide interest since Darwin's time, but controversies have arisen regarding the effect of selection pressure on mutation, and different models have been proposed in the meantime [150]. It is now believed that it is impossible to separate any form of mutation from the effect of selection. In order to investigate mutation pathways Roth and Andersson [149] define *adaptive mutations* as fitter mutations that arise under selective conditions. In subsequent work [71, 96, 163], they propose a gene duplication-amplification model to study the mutagenesis stimulated by enhancement of selection. In addition, a recent study by Weinreich et al. [180] on the effects of Darwinian selection on random mutation argues that environmental selection can make some multi-step mutation pathways inaccessible. Specifically, by studying "five point mutations" in a lactamase allele that can increase bacterial resistance to an antibiotic, several pathways are in principle possible for these successive mutations. After calculating the different probabilities of these pathways, their experimental results show that under intramolecular interactions only a small number of pathways are really accessible. This is quite an interesting result because mutations might be channeled by some unknown fitness-increasing principle(s) and the resulting proteins might be reproducible and even predictable. These feedback and interaction mechanisms may reduce the harm that mutations could bring to an organism. This point of view also conforms to Kirschner and Gerhart's definition of evolvability [88], which they define

as “the ability to reduce the potential deleterious mutations and the ability to reduce the number of mutations needed to produce phenotypically novel traits”. If mutations can be channeled, fewer changes might be needed to generate a required adaptation and, therefore, evolvability would be improved by this reduction in cost of mutations.

In EC, mutation is regarded as an important exploratory operator. Artificial evolutionary search should be good at both exploring suitable genetic novelty and maintaining successive improvements. Holland [73] discusses this principle as the tension between “exploration” and “exploitation”. The mutation rate is important to keep this balance, and it has already been studied as an evolvable parameter contributing to evolvability. Bedau et al. [21, 22] divide evolutionary adaptation conceptually into two stages: the novelty stage, where an evolving system enhances its adaptability against a changing environment, and the memory stage, where the evolving system is building up this adaptability through incremental improvements. By providing a simple two-dimensional model, Bedau et al. postulate that the mutation rate should increase during the novelty stage and decrease during the memory stage. This fluctuation of mutation rates is able to keep the balance between evolutionary novelty and memory, and thus increases the evolvability of adaptive systems.

However, compared to natural evolutionary systems, genotypic variation in computational systems is somewhat arbitrary and not as adaptive. First, the fixation process of mutations is not simulated in most EC algorithms, because all changes to individual sequences are mostly translated into phenotypic properties. Recovery or repair mechanisms are usually not applied to individuals suffering deleterious mutations, which make those individuals unfavored during the selection process. Second, the selection-driven mutation pathways found in natural systems are an interesting di-

rection to explore for computational models and may be considered in future research in EC.

Gene duplication

Gene duplication is an important mechanism creating new genes and new genetic subsystems. This mechanism has been recognized to generate abundant genetic material and contributes substantially to biological evolution [130]. A large number of duplicate genes have been discovered to exist in vertebrate genomes [123], and a repeated number of whole genome duplications has been established as key events in evolutionary history [166]. In modern Biology, gene duplication and its subsequent function-specialized divergence are widely believed to be a major reason for functional novelty.

Once a gene duplication has occurred a complex fixation process on the duplicate genes takes place. *Purifying selection* and *gene conversion* are the main pressures affecting the survival of duplicate genes [125]. Many duplicate genes become pseudogenes after one or more mutations disable them and no promoting function is yielded. However, multiple copies of identical genes can, after duplication, promote functional redundancy against fatal changes. The process of *pseudogenization* is reported to occur in the early stages of a rapid evolution process [68], with evidence of many pseudogenes found in the human genome. Other duplicate genes are changed by selection pressure and functional divergence. Subfunctionalization and neofunctionalization are the two main mechanisms of functional divergence [191]. In *subfunctionalization* of two gene duplicates, each copy adopts a different aspect of the function of the original gene. Both copies will be stably maintained because both aspects of the

function are indispensable. Subfunctionalization leads to functional specialization by dividing multi-functional genes once the newly emerged genes perform better. Alternatively, some relatively new function can also evolve after gene duplication [192], and this process is called *neofunctionalization*. This has been termed the Dykhuizen-Hartl Effect [38] earlier, where a random mutation is preserved in the duplicated gene by reducing selection pressure due to functional redundancy that results from gene duplication. Such mutations may accumulate and induce a genetic function change depending on conditions of the (dynamic) environment. New adaptive functions may thus be generated and later preserved during evolution. By possibly creating novel functions and allowing evolution under fewer constraints, neofunctionalization is an important consequence of gene duplication.

The mechanism of gene duplication contributes substantially to genomic and organismal evolution. It provides abundant material for mutation and selection, and allows to specialize functions or generate completely new functions. The acceleration of protein sequence evolution after gene duplication has recently been reported in research on yeast genes by Gu et al. [66]. The authors use an additive expression distance between duplicate genes to measure the rate of expression divergence, and rapid evolution of gene expression and regulatory divergence after gene duplication is observed.

In summary, the mechanism of gene duplication can considerably increase evolvability of a system by reducing the cost of mutations. In EC, the idea of using gene duplication and deletion operators was proposed some time ago. Those operators are in general based on the method of variable-length genotypes, and are executed with predefined duplication or deletion probability [65, 93, 154, 155, 181]. Unfortunately,

so far only application-oriented work has appeared with different representations [26], and a common framework for this concept is missing. More details of gene duplication in Biology should be taken into account to benefit computational evolution. In particular, the question of how gene duplication reduces the limitations of mutation and selection, and in the process promotes evolvability needs to be studied. Is there a way to implement functional specialization and innovation through gene duplication in EC?

Point mutation vs. gene rearrangement

A point mutation occurs when a base on a DNA sequence is changed into another base at the same locus. Gene rearrangement is a change in the order of a DNA sequence on a chromosome. This change can be an inversion, translocation, addition, or deletion of genes. Earlier research focused mostly on *Single Nucleotide Polymorphisms* (SNPs) in genomes due to the enormous complexity of genetic sequence analysis, but gene rearrangements have always been believed to contribute to evolvability, possibly even more than simple point mutations [87]. Recent development of technology has now facilitated the shift in focus from a locus-based analysis to a genome-wide assessment of genotypic variation [53].

Genetic rearrangements rather than point mutations can maintain the connective information carried by gene sequences. Because genes form networks of functional control, rearrangement is better able to preserve internal structures.

The ubiquity of *Copy Number Variations* (CNVs) has been realized recently in mammalian genomes by different groups of biologists, such as Iafrate et al. [79], Sebat et al. [157], and Tuzun et al. [171]. CNV is regarded as a predominant type of

genotypic variation leading to vast phenotypic diversity in mammalia. CNVs show that large segments of DNA, with sizes from thousand to millions of base pairs, can vary in copy number of genes. This variation can lead to protein dosage differentiation in the expression of genes, and CNV is therefore regarded as being responsible for a significant proportion of phenotypic variation [53]. The mechanisms that create CNV have not yet been clearly understood, but some hypotheses have been proposed in the literature. Fredman et al. [52] and Shaw and Lupski [158] propose that CNV might be the result of large segmental gene duplications or non-homologous recombination events.

Recent bioinformatics research uses statistical and computational tools to analyze chromosomal evolution by a comparison of genome-rearrangements between sequences of related species [152]. Although the biochemical mechanisms of gene rearrangement are still far from being fully understood, we believe it is time to start using such rearrangement operations in computational models in EC. Particularly, the recent discovery of CNVs requires attention by computer scientists, in order to achieve similar benefits in EC.

2.2.2.2 Phenotypic variation

As mentioned in the previous section, despite their vast phenotypic differences, metabolic processes and cell structures in bacteria and humans are quite similar [89]. What, then, makes humans so different morphologically from other organisms? It is phenotypic variation. Unfortunately, the relation between genotypic variation and phenotypic variation is still not fully clarified in current biological opinion. Since selection acts on phenotypes rather than on genotypes, phenotypic variation should be used

to explain the immense diversity among organisms. Here, we discuss several aspects of phenotypic variation. We leave the discussion of the mapping process between genotype and phenotype that controls the direction of phenotypic changes resulting from genotypic variation to Section 2.2.4.1.

Conservation and relaxation

According to Kirschner and Gerhart, evolution possesses two important features: conservation at the molecular level and relaxation at the anatomical and physiological level [89]. By conservation it is meant that the genetic components of organisms tend to maintain relatively stable structures; relaxation refers to the less constrained phenotypic diversification of organisms. The authors state that conservation on the genotypic level reduces the constraints on the phenotypic level.

In Darwin's evolutionary theory, all organisms have evolved from the same ancestor. After primal initialization and evolution, genetic structures of organisms are highly conserved during the course of billions of years [178]. This can well explain why the number of human genes is only a few times that of bacterial genes but significant anatomical and behavioral differences exist between them. The surprisingly small number of genes in humans and other complex organisms demonstrate that the great diversity and complexity at the anatomical and physiological levels have to rely on and organize/reuse limited genetic material. When certain organisms improve their adaptivity in order to survive in a new environment, the regulation system only has to recombine existing mechanisms for the generation of adaptive functions, which requires little or no new genetic material [88]. Not only are gene sequences highly conserved, the *core processes* of coordination of the genetic material are also well con-

served from the time they initially emerged [89]. These conserved core processes are used repeatedly for different purposes and functions under different circumstances, at different times, with different genetic material. The Baldwin Effect [162] explains that phenotypic variation is not generated out of the blue but through regulation of existing components in organisms: Mutation simply stabilizes and extends what has already existed to improve somatic adaptability towards external stimulations.

This conservation mechanism can efficiently prevent lethal changes in genotypes and is an economic method to increase the adaptability of organisms. New material is not needed to adapt to changing environments, but few modifications will suffice.

Functional innovation is heavily constrained due to molecular interactions among various genetic components that are involved to produce a specific trait. If the participation of more genetic components is needed, it becomes harder for functions to change. In fact, relatively little genetic material is required to generate all proteins of organisms. Under selection pressure from a changing environment, organisms have to yield adaptive phenotypic traits to survive, however, and the highly conserved core processes mentioned above are used repeatedly to generate new cooperation among the conserved genetic material, bringing about fitter function and behavior. Relying more on the combinatorics of components is equivalent to relaxing phenotypic variation.

The relaxation on phenotypic variation has been highlighted as the notion of “deconstraint” in Kirschner and Gerhart’s [88] research on evolvability which studies the mapping from genotype to phenotype. Enhancing phenotypic variability under changing environmental conditions allows nature more evolvability. Not only can deleterious changes be avoided, but nonlethal genetic and phenotypic variation is

indeed the material from which innovation can be generated.

Turning again to EC: What is the role of conservation and relaxation in EC? First, an economic use of genomes or building blocks can help to conserve genetic information. Second, it can be assumed that by reducing the constraints on changes to a phenotype the exploratory capability of a computational system to find better solutions can be enhanced. How such a process can be implemented in actual systems is presently unknown, but a worthwhile line of inquiry.

Modularity

Modularity is a widespread structural property of complex systems. It has attracted considerable interest from studies of both natural and artificial evolutionary systems, and is regarded as strongly related to evolvability [177] and the acceleration of evolution [161].

Modularity exists at various levels, e.g., at the level of gene expression or embryonic development. Here we adopt the definition of modularity proposed by Simon [160] in his research on hierarchies in complex systems.

Definition 4: Modularity. In a complex system, modularity refers to the property that a loose horizontal coupling exists between the entities at the same level of this system. [160].

Simon [159, 161] further defines that “a system is *nearly decomposable* if it consists of a hierarchy of components, such that, at any level of the hierarchy, the rate of interaction within components at that level is much higher than the rates of interactions between different components”. Although this “Near Decomposability (ND)” is

attributed to a *vertical* separation while modularity describes the separable property of components horizontally at the same level, they seem closely related in that they both describe how a complex system is decomposed into sub-systems.

The modularity of genotype-phenotype mappings has been extensively studied in gene expression. It reduces harmful pleiotropic effects of gene expression and can lead to adaptive phenotypic variation. *Pleiotropy* is a general property of genotypic variation, where one change at the genetic level can cause a multitude of functional changes at the phenotypic level. Pleiotropy can generate both advantageous and disadvantageous results. Pleiotropy can sometimes generate unexpectedly improved functions, but can also be harmful or even fatal to evolutionary systems [6]. Since a gene can affect multiple functions, optimizing one particular function at the phenotypic level inevitably incurs side-effects on other functions. Bonner [25] proposes the notion of “gene nets” by grouping gene actions and their products into discrete units during evolution. In general, for a given organism, the mapping from genotype to phenotype can be divided into modules such that the sets of genes in one module only affect the functions in that same module. The mapping is therefore decomposed into groups of independent “sub-mappings”. Bonner finds that the phenomenon of gene nets becomes increasingly prevalent as organisms become more complex. Wagner and Altenberg [177] investigate modularity in genotype-phenotype mappings from both perspectives, Biology and EC. They interpret modularity as a means for dividing phenotypic traits into different “compartments” to reduce interference among different optimization modules. With such modularity, optimization of a function in one module has no effect on functions in other modules. As a result, pleiotropy can be confined to a known set of functions during evolution.

Wagner and Altenberg [177] further propose that modularity results from evolutionary modifications in natural organisms. In their view, the evolution of modularity follows two mechanisms, dissociation and integration. Dissociation is the suppression of pleiotropic effects by disconnecting interactions between different modules, while integration is realized by strengthening of pleiotropic connections among traits in the same modules. Both mechanisms are driven by selection pressure.

Thus, modularity can be conceptualized as an evolutionary mechanism to promote evolvability. It reduces the interdependence of disjoint components and consequently reduces the chance of pleiotropic damage by mutation [88]. It allows genotypic variation and selection to affect separate features in a complex system and to evolve various functions without interference [101]. Sub-systems as part of an entire system can evolve faster to optimize their local sub-functions individually, by decreasing crosstalk between genetic changes. In a study of encoding schemes in EC by Kazadi et al. [86], a *compartment* is defined similar to a module in the genotype-phenotype mapping, and such compartmentalization at different levels is claimed to contribute to the acceleration of evolution. In RNA research, Manrubia and Briones [109] propose that the increase of molecule length and subsequent increase in functional complexity could be mediated by modular evolution. They find that short replicating RNA sequences with a small population size can be assembled in a modular way and can create complex multi-functional molecules faster than conventional evolution of complex individuals toward multiple optima.

Modularity in general has been widely used in computer science and engineering by subdividing complex entities into smaller components to yield higher computational efficiency, and we expect it to play a major role in EC. In current EC models,

phenotypic variation is mostly generated from genotypic variation with mappings that are not very complex. It is our opinion that considering genotypic and phenotypic variation separately but connected through a number of complex and sophisticated evolutionary mechanisms will allow EC to benefit substantially.

2.2.2.3 Epigenetic mechanism

Epigenetics has become a new research direction in evolutionary biology [80]. Literally, “epi”-genetic control lies in the regulation of gene expression *without* changing the DNA sequence itself, so it is “beyond the conventional genetic” control. Epigenetic regulation arises during the processes of organism development and cell proliferation, triggered by intrinsic signals or environmental stimulations [82]. Epigenetic changes are heritable in the short term from cell generation to cell generation, and these stable alterations do not involve mutations on DNA sequences. Epigenetic regulation of DNA expression lies at the heart of many complex and long-term human diseases [17].

Previous research in genetics mostly focused on the sequential information carried by DNA. However, DNA sequences are coiled up in cells in intimate complexes with the help of so called histone proteins. A DNA sequence wrapped with histones comprises a *nucleosome*. *Chromatin* is the complex of nucleosomes in the nucleus of cells which participates in the control process of gene expression. The chromatin composition varies according to cell type and response to internal and external signals. The different composition of chromatin may affect expression and thus change the produced proteins even in the absences of DNA sequence modification [3].

The main mechanisms of epigenetic control are DNA methylation and histone modification [82]. Modifications to chromatin, either on the DNA sequence itself

(*DNA methylation*) or on its surrounding proteins (*histone modification*), affect gene expression and can be inherited from cell generation to cell generation during cell division. DNA methylation is a chemical addition to DNA sequences. Genes with methyl marks are repressed in expression, despite their unchanged DNA content [182]. In histone modification, the tails of histone proteins are modified by different molecular attachments, e.g., acetyl, phosphoryl, and methyl groups. If acetyl groups are attached to the histone tails of a chromatin, it will be loosely packed, a state called *euchromatin*. In euchromatin, DNA is readable and can be transcribed into RNA and later translated into proteins. In contrast, if methyl groups are attached to histone tails, chromatin is tightly compressed, a state called *heterochromatin*. In the heterochromatin state, genes are inaccessible to the transcriptional machinery such as RNA polymerase or to transcription factors, and genes are prevented from being transcribed [64]. Other mechanisms recognized to be responsible for epigenetic regulation of gene expression include chromatin remodeling, histone variant composition, and non-coding RNA regulation. A discussion of these mechanisms can be found in Allis et al. [2].

The key feature of epigenetic mechanisms is to coordinate internal and environmental signals which can collaborate to modify protein production [82]. The underlying interactions involve various molecules, such as DNA, RNA, and proteins, but the extensive feedback between these molecules is still beyond our current understanding.

We believe that epigenetics opens up a new field in evolvability studies for both Biology and EC. Sophisticated epigenetic feedback networks suggest a new structure for EC compared to the linear flow of computation usually employed in the literature. For instance, in dynamic optimization problems, not all genes responsible

for different subfunctions need to be expressed all the time. We anticipate that a “controller switch” can be integrated into the genotype allowing short-term changes, where fragments of the genome can be turned on and off in response to feedback from the outside. Such a mechanism for repression of expression has barely been used in computation. Similar multi-layer adaptive encoding schemes have been proposed, e.g., the *messy Genetic Algorithm* (mGA) [62] that combines short building blocks to form variable-length chromosomes to increasingly cover all features of a problem, or *diploid Genetic Algorithm*, e.g., [172] using a two-chromosome representation to adapt phenotypic variation in dynamic environments. However, existing work has not embedded the organizational epigenetic control in algorithms that would allow significant flexibility in changing environments. We anticipate that epigenetic mechanisms will play a crucial role in increasing the evolvability of EC algorithms.

2.2.3 Stabilization operations

There are two main stabilization operations in evolution: recombination and selection, which will be reviewed in this section.

2.2.3.1 Recombination

Recombination is a process that generates combinations of existing genetic material in contrast to mutation which creates new alleles. Recombination is regarded as an important force shaping genomes and phenotypes. Since some highly efficient and accurate computational methods can be used in Biology, analysis of gene recombination has made much progress by way of comparing aligned genome sequences. These comparisons facilitate a better understanding of several aspects of genetic and

evolutionary biology, notably genotypic and phenotypic variation and genome structures [140].

Recombination exchanges genetic material between two DNA sequences swapping strands between one or multiple crossover points. Recombination can occur on homologous or non-homologous sequences. The former is more prominent in research because it is an integral part of the cell style. Generally, research on recombination focuses on prevalent eukaryotic organisms rather than prokaryotes, which do not have sex. Unequal crossover is quite common and may lead to duplication or loss of some genes (discussed in Section 2.2.2.1) and other results [164]. Recombination events can take place between different gene sequences, as in *intergenic recombination*, or between alleles on the same gene sequence, as in *intragenic recombination* [140]. Despite various forms of recombination, their outcome is crossover at one or multiple points and a swapping of fragments of genetic sequences.

The rate of recombination (usually regarded as the frequency that recombination happens on certain portions of sequences) can significantly affect the rate of adaptation. It is usually higher than the rate of mutation, which implies that recombination introduces much less lethality to an evolutionary population than mutation. Instead, it advances evolution remarkably by stabilizing adaptive traits from parents to offspring. By drawing a recombination map of the human genome, Kong et al. [90] discovered that recombination rates vary in different regions of the genome. This variation is due to such functional features as gene density, other gene properties, and frequency of sequence repetitions. Recombination rates are also different in autosomes between different sexes. Recombination contributes to producing both genotypic and phenotypic variation, and is able to repair DNA double strand breaks.

In EC, recombination operations are considered an essential search strategy. Chromosome coding is much more flexible in computation than in nature, and thus, various recombination techniques have been proposed and studied, including double-parent and multi-parent crossover [41], fixed-length chromosome and variable-length chromosome crossover [62, 69], and homologous and non-homologous crossover [98, 127, 138]. High recombination rates are usually also adopted in computation because of its perceived efficiency in generating beneficial genetic and phenotypic variation. Elsewhere, adaptive recombination rates are proposed to strike a balance between exploration and exploitation [167]. In most of these adaptive recombination rate schemes, modification of recombination rates is based on fitness value. Different from natural recombination mechanisms, most adaptive recombination rate proposals simply react to the current status of the search, in order to escape from local optima. However, rate adaptation in Biology is much more complex and suggests other models for computation. For instance, the rate may vary among different individuals or in different modules serving sub-functions in the genome. Such function-specific recombination rates could also consider the method of “compartmentalization” for modularity (Section 2.2.2.2). The notion of *epistatic clustering* in contributing to evolution of evolvability has recently been studied [137]. That is, genetic linkage patterns between different loci are claimed to affect recombination rates, and the simultaneous optimization of different recombination rates on different traits would be realized by a method called epistatic clustering. Evolvability would be improved through co-evolution of trait clustering and recombination mechanisms.

2.2.3.2 Selection

Although Darwin's theory of evolution being directed primarily by natural selection has been the subject of much debate, selection is an extremely important operation to stabilize the functional traits already generated by some exploratory operations [124]. The balance between selection and diversity of an evolutionary population has been a critical problem, and the dynamic pressure and some consequences of selection are still under active investigation.

Environmental selection originates in the external surroundings and enforces the adaptivity of organisms to survive and reproduce. Darwin environmental selection has received extensive attention in evolutionary biology. Natural selection is extremely important for adaptive evolution in natural populations [72].

Selection can act at different levels depending on its targets [88]. These might be individual selection, individual-and-population selection, or population selection. At the individual level, fewer mutational changes are required for a new adaptive trait. An individual can also interact with others in a population, such as through recombination, and survive under selective pressure as a member in this population. At the highest level, selection can happen on the level of an entire population given large environmental impact, and the entire population can, as a whole, escape from extinction. Theoretically, some small groups of the lineage might go extinct, but the entire line will be able to survive even if it might have to go through population bottlenecks.

Environmental selection is widely accepted as contributing significantly to natural evolution, and selection has entered as the mainstream of studies in evolvability. As

a potential to generate adaptation, evolvability is difficult to observe and to select for. However, there is increasing research arguing that evolvability is selectable and environmental selection can improve the evolution of evolvability. In the real world, the environment is changing constantly and results in changes in beneficial mutations, and there is a growing acceptance that a changing environment is a key ingredient to studying evolvability. Selection pressure is a critical operator to control an evolutionary process. Earl and Deem [39] suggest that selection pressure is increasingly strong when the environment becomes uncertain. Dramatic environmental changes lead to selection for better evolvability. They consider evolvability as a selectable trait, and facilitating environmental changes can be a method to accelerate evolution. A recent simulation by Kashtan et al. [85] in a biologically realistic setting also suggests that varying environments may accelerate natural evolution. In their work, different scenarios of temporarily changing optima were used. Kashtan and Alon [84] report using a simulated evolution system that a goal that varies in a modular way can speed up evolution. Other work [30] takes into account the effect of the rate of environmental change. By observing the dynamics of *adaptive walks* under scenarios of varying speeds, they find that environments with varying rates of change have noticeably different effects on the fixation of beneficial mutations, the substitution time required, and the final phenotypic variation.

In EC, selection strategies can affect the search capability of an algorithm significantly. Different selection strategies have been proposed and the dynamics of selection pressure has been studied extensively [24, 60]. Since the effects of environmental selection on the evolution of evolvability have been recognized, further research on the dynamics of selection is required. Moreover, somatic adaptation should be considered

when applying selection. Group-based selection methods should also be studied for varying selection pressure, so that a balance between the development of a minority and of the entire population can be dynamically achieved.

2.2.4 Quality differentiation

As discussed previously, exploration and stabilization processes generate variation and adaptation at different levels. In contrast, the process of quality differentiation quantifies these adaptations and to distinguish between individuals. Two aspects are involved in quality differentiation, the genotype-phenotype mapping and fitness evaluation. Genotype-phenotype mapping translates genetic information into visible functional phenotypes, and fitness evaluation measures the adaptation of individual variants based on their ability to survive and reproduce.

2.2.4.1 Genotype-phenotype mapping

In EC, mapping from genotype to phenotype is often an encoding process, especially in evolutionary algorithms and evolutionary strategies, where the mapping mechanism is simply to calculate a fitness function of each individual. However, in nature, the mapping process is much more complex, typically from highly conserved genotypic information to greatly divergent polymorphism in phenotypes. The fundamental process in biological genotype-phenotype mapping is gene expression, and the most important mechanism in this process is regulation of gene expression, which will be discussed next. Since research on transcriptional regulation has discovered increasing evidence that RNA plays an important role in gene expression, the transcriptome, i.e., the set of all transcribed RNAs, will be reviewed then.

Regulation of gene expression

In Biology, the core processes (Section 2.2.2.2) of organisms are responsible for generating anatomy and behavior using genetic and cell materials. These core processes include metabolism, gene expression, and interaction among molecules and cells [89], which are well-conserved but still under exploration. Regulation of gene expression is the most important mechanism among the core processes to facilitate organismal novelties in evolution. Kirschner and Gerhart highlight the characteristics of “conservation” and “economy” in regulatory core processes in [89].

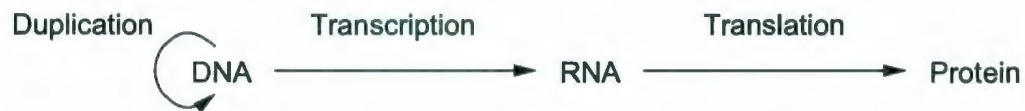


Figure 2.2: The Central Dogma of Molecular Biology.

Scientists have made considerable progress in to understand the process of gene expression for decades. In 1956, Crick proposed the *Central Dogma* of molecular biology, as shown in Figure 2.2, which describes the one-way transmission from DNA to protein. The circular arrow around DNA symbolizes that a DNA is a template for self-replication. The arrow from DNA to RNA indicates that an RNA is transcribed on a DNA template, and the arrow from RNA to protein signifies that a protein is translated on an RNA template.

Subsequent biological research revealed that the process of gene expression is much more complex than such a linear flow, and involves a considerable number of complex regulation operations. The Central Dogma was challenged by discoveries of proteins

playing an important role in regulation of gene expression, and most recently, the non-coding RNA control of chromosome architecture proposed by Mattick [113]. In this section, we concentrate on gene expression regulation by proteins and will discuss RNA effects in the next section.

Recall the discussion of genome redundancy in Section 2.2.1. Coding regions on genetic sequences that can be expressed into proteins only occupy a small portion of the entire genome in eukaryotic cells. This discovery indicated that a huge number of regulatory elements exist in genomes that participate in generating adaptation in evolution according to changes in environments. Although living systems have evolved for billions of years, regulatory core processes in various organisms have remained mostly unchanged despite species divergence. By comparison of related species from the same ancestors, such as humans and chimpanzees, at both the molecular and organismal levels King and Wilson [87] had already found in 1975 that genetic structures in these two species are almost the same; while at the organismal level, the anatomy, physiology, behavior and ecology of these two species are significantly different. This suggested to them that the complex adaptive evolution is produced by a combination and multiple utilization of similar, highly conserved genetic components under the control of regulatory systems.

A key step in the regulation of gene expression is transcription. Studies there are concentrated on two primary components: *promoters* and *transcription factors*. Promoters, also known as cis-regulatory sequences, are responsible for regulatory transcription. Cis-regulatory sequences are a part of non-coding DNA sequences, and they can determine the target genes and the length of the loci that will be transcribed under which conditions. Transcription factors are proteins interacting

with these cis-regulatory sequences by binding to certain sites on DNA sequences. Readers Interested in more details are referred to Wray et al. [185]. Transcription factors act either as activators or as repressors of gene expression. For example, if a transcription factor A binds to a site on a DNA sequence that is responsible for generating protein B, then this factor A is regarded as a repressor to protein B. In addition, as a protein itself, factor A also has its template gene sequence. If another transcription factor C can bind to this site and repress the generation of protein A, C acts as a repressor to A but in turn as an activator to the expression of protein B. These activators and repressors can work together as a network of logic control. Promoters usually contain a number of binding sites for transcription factors, where each site can only be occupied by one factor at a time. These binding sites occupy, however, only a small fraction of sequences, and are distributed unevenly. Some binding sites of different functions can overlap. Furthermore, binding affinities of different materials are important for regulation as well. On the other hand, most transcription factors have numerous target genes and use priorities in binding with any of them [185]. This sophisticated network endows the regulation system with the high robustness and plasticity necessary for evolution of organisms.

Evolution of cis-regulatory sequences as non-coding sequences is considerably different from that of protein-coding sequences, and is less understood. King and Wilson [87] suggest that protein-coding sequences are highly conserved during evolution. It is mutations on promoters that causes most morphological variation. Research on the evolution of transcriptional regulation has become mainstream in molecular biology in recent years [185]. In particular, Roderiguez-Trelles et al. [148] find that significant substitution rate differences exist among different promoters, and even

some neighboring cis-regulatory promoters involved in the same regulatory network can have different evolution rates. Moreover, Stone and Wray [168] propose that local point mutations on binding sites can lead to rapid evolution in gene expression, which indicates their potential of accelerating evolution. Wagner [173] points out that other simple changes such as gene duplication and deletion of promoters can also result in rapid evolution in gene regulatory networks. By comparing genomes, Fondon and Garner [50] discover that gene-associated tandem repeat expansions and contractions exist and give rise to rapid morphological evolution. In their experimental research, a tandem repeat mutation shows both elevated purity and intensive length polymorphism among different dog breeds. Mutations on non-coding sequences can modify regulation of the target genes, the length of coding loci to transcribe, and the occurrence conditions. Furthermore, they also result in morphological variation and accelerated phenotypic evolution.

Since the mechanisms of regulation of gene expression can well explain many phenomena in evolvability and rapid evolution in living systems, research on artificial regulatory networks has now started in computer science. Several models of artificial evolution regulatory networks have been proposed such as Banzhaf et al. [11, 12, 14, 97], Chavoya and Duthen [28], Mattiussi and Floreano [115], Nehaniv [122], etc. These artificial models intend to generate regulatory behavior akin to that of natural systems. However, these research efforts are still in their early stages, and more work on evolvability and dynamics in artificial regulatory networks is necessary.

Transcriptome

The transcriptome, or collection of transcripts, refers to all RNAs produced in a single or a group of cells, working as an intermediate component of gene expression. In high-level eukaryotes such as humans, most regions of the genome are not transcribed, and most regions of the transcriptome are not translated into protein. What necessitates the existence of such a large number of RNAs in the transcriptome of high-level eukaryotes? Regulatory functions is one answer to this question. Although regulation of gene expression starts with the transcription step, these transcribed but non translated sequences or non-coding RNA sequences act as regulators for translation in gene expression, and currently attract increasing interest in biological research [54, 114].

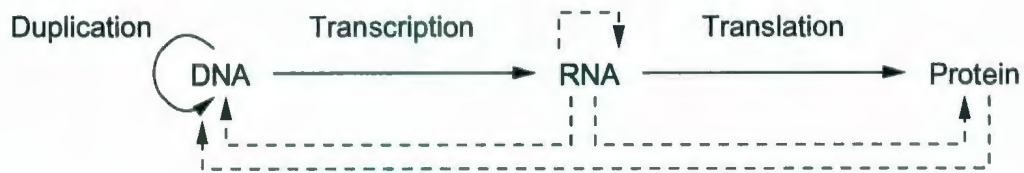


Figure 2.3: Eukaryotic genetic system.

An RNA is not just a temporary medium between genes and proteins for a one-way information flow as described in the Central Dogma. In high-level eukaryotes, the information transmission from DNA to protein is not a one-way process, but involves many functionalities of the transcriptome. The new perspective of gene expression proposed by Mattick [113, 114] can be seen in Figure 2.3. Compared to a prokaryotic genetic system, a eukaryotic system has a parallel control mechanism with multiple outputs and information transfers. Rather than a simple medium of gene expression, RNA metabolism and interaction have been discovered as playing an important role

in gene expression regulation.

Mattick [112] proposes that non-coding RNAs participate extensively in gene expression regulation, being present in about 98% of all transcriptional outputs in eukaryotes. In research on the human transcriptome, Frith et al. [54] found that non-coding RNAs play an important role in generating phenotypic variation. Non-coding RNAs can be classified into two categories: introns and other non-coding RNAs.

Regulation of the transcriptome shows contributions to evolvability and rapid evolution. Introns, an important category of non-coding RNAs, are found to be more susceptible to mutations than their neighboring protein-coding exons. Rather than having no function, as thought previously, the fewer constraints on introns offer flexibility to generate new functions and rapid protein sequence evolution during the process of regulation, especially in connection with alternative splicing. The evolution of RNA communication networks may also accelerate the evolution of gene expression, as observed by Mattick [112]. These RNA communication networks, which describe interaction among different layers of RNA signaling, provide a sophisticated regulatory architecture, enabling DNA-DNA, DNA-RNA, or RNA-RNA communication.

Compared to natural systems, the genotype-phenotype mapping in EC is rather primitive still and a transcriptome is mostly missing in algorithms. The complex RNA parallel information transfer framework inspires various applications. Based on what computational models have already achieved with artificial regulatory networks, more mechanisms should be implemented, especially the newly discovered powerful mechanisms of transcriptome regulation.

2.2.4.2 Fitness evaluation

Fitness evaluation measures behavior or function of individuals. In nature, fitness of an individual or species is implicit and subject to natural selection, whereas in EC, fitness is mostly based on numerical values of an individual as a solution to a given problem, and this fitness is explicit.

In nature, adaptable species survive by passing different challenges, and less adapted species may become extinct during evolution. Adaptability lies not only in the currently existing adaptivity to the environment, but also in the capability to generate more adapted offspring. In essence, fitness of natural organisms is implicit and is subject to natural selection. Empirically, biologists use mathematical methods to quantify fitness. *Individual fitness* usually refers to the viability of an individual, i.e., its probability to survive [57]. Moreover, individuals having more offspring can be considered as fitter ones since their genetic information is more likely to be preserved. Other than at the individual level, in classic population genetics literature [33], the *genotype fitness* quantifies the frequency changes of a genotype in a population during transformation from one generation to the next. Various measures have been proposed in the biological literature (see [134, 165] for detailed reviews).

The above implicit fitness in natural organisms emphasizes evolvability under intricate pressures from interactions among evolutionary components, internal or external to these organisms during a long, continuing evolutionary process [137]. In reality, the fitness of individuals in a system can vary a great deal. Moreover, a large-scale quality differentiation exists in almost every natural evolutionary system, and these vastly diverse evolution systems exhibit substantial evolvability. Since selection and

evaluation act directly on observable phenotypic functions but evolvability only provides the potential for better functions, selection and evaluation for evolvability are not observable directly.

Since EC has been widely applied in many areas of industry and academia, fitness evaluation arises as a difficult problem because it is usually very CPU-intensive. In the current literature, two main methods of fitness evaluation are employed, absolute fitness and relative fitness. Absolute fitness of each individual usually refers to its value of a specified fitness function. Relative fitness compares different individuals and gives a rank to each individual to produce a record of winners. This latter method is good at suppressing exceptionally good individuals, thus, helping an evolutionary system to escape from premature convergence. In fact, evaluating the fitness of each individual is usually difficult for many optimization problems in the real world because explicit fitness can be hard to define and expensive to calculate. As a result, *fitness approximation* has been proposed with differing levels of approximation, including “problem approximation”, “functional approximation” and “evolutionary approximation”. Jin [83] has surveyed these approaches. They are sensitive to training data and to varying constraints of different models, so a common framework would be required. Moreover, Reisinger and Miikkulainen [145] propose an evolvable representation and an evaluation strategy to exert indirect selection pressure on evolvability. In their work, a systematically changing fitness function is adopted according to a special evolvable representation that can reflect efficiently how genetic changes restructure phenotypic variation. Thus, evolvability can be evaluated through the way such a systematic structure can expand in phenotypes. These approaches might provide a good starting point to simulate the implicit adaptive fitness

evaluation from nature, a method that has good prospects for detecting evolvability in EC.

2.3 Discussion

Since Darwin proposed his theory of natural evolution based on heritable variation and natural selection, numerous research efforts have been dedicated on this subject. In modern Biology, many details about mechanisms of evolution and factors that can affect evolution have been revealed. Besides understanding the history of evolution, biologists are currently paying more attention to the capability of organisms to evolve and to the evolution of such capability in the open-ended natural evolutionary process. Meanwhile, in research of artificial evolutionary systems, researchers also work on improving the capability of computational models of evolution by studying more intelligent and adaptive mechanisms.

Evolvability, as the capability to evolve, has received considerable interests in recent research in both Biology and EC. Substantial work has been done on this topic in both disciplines, and many factors are found to contribute to evolvability.

In this chapter, we started from notions and new discoveries in Biology, including aspects that have been recognized to improve evolvability in natural systems. The order in the presentation of these ideas conforms to the flow of an evolutionary algorithm. In each part of the flow, we first reviewed relevant results in Biology. Next, a brief survey was provided to describe current research status in EC, followed by an outlook for further research. Our goal was to describe and present new research outcomes to computer scientists, especially in the field of EC. Since it is accepted

that artificial evolutionary systems are much less evolvable than natural systems, we hope these ideas can inspire new methods and applications in EC.

Chapter 3

Rate of Evolution

Related to the theme of evolvability is the rate of evolution. Evolvability defines how likely a system can generate adaptive phenotypic variation and the rate of evolution describes how fast this evolutionary process can proceed. Evolvability indicates the *potential* of a system to evolve, thus, it is difficult to capture and quantify. Rate of evolution provides an observation on the “outcome” of evolvability, and it can be an important tool to study evolvability. Therefore, evolvability and rate of evolution are interrelated and crucial aspects in both Biology and EC.

As a fascinating topic in evolutionary biology, the rate of evolution has caused debates already since Darwin’s time. Darwin himself held the view of *phyletic gradualism*, hypothesizing that most evolution occurs uniformly, gradually molded by selective conditions. Others were of a different opinion, and Eldredge and Gould [44] proposed the theory of *punctuated equilibria*. According to this idea, evolution occurs through bursts of innovation followed by long periods of stasis, a major challenge to Darwin’s orthodoxy.

3.1 Measuring rate of evolution

The rate of evolution is understood somewhat differently in EC and in Biology. The goal of evolution is much more explicit in computational systems than in nature. It is to find the fittest solution to a given problem. Therefore, the rate of evolution in EC usually refers to how fast an EC population improves its fitness value, i.e., the *rate of fitness progression*. The ability to define explicit phenotypic fitness is one of the most distinguishing features that differentiate EC from natural evolution. In order to investigate the performance of a computational model, the rate of evolution is thus mostly measured by the speed of fitness function improvements. Other ad hoc methods are also utilized in EC, like the efficiency of algorithms and CPU time.

There are, however, some methods at a deeper level than simple fitness function improvement, that can be found in the literature. Bedau and Packard [20], for instance, propose a method for visualizing evolutionary adaptation. This method is useful to identify and measure the capability of creating adaptation during evolutionary processes. It is based on calculating evolutionary activity statistics of components in an evolutionary system. During a decade of extensive development, the notion of *evolutionary activity* has been applied to various scales of genetic components, including alleles, allele tokens, phenotypic equivalence classes of alleles and whole genotypes, in both artificial evolutionary systems and the biosphere. In their later work, two aspects for evolutionary adaptation were emphasized: the *extent* and the *intensity* of evolutionary activity [19, 142]. The extent of evolutionary activity concentrates on how much of an adaptive structure is present in an evolutionary system, while the intensity concerns the capability of generating new adaptive structures. The measures

of *cumulative evolutionary activity* and *mean cumulative evolutionary activity* characterize the extent of a system's evolutionary adaptation. In addition, *new activity* is a measure of the intensity of a system's evolutionary adaptation. Evolutionary activity can be quantified and visualized during evolutionary adaptation. Its derivative is the concentration of a component's current presence, and its second derivative can be argued to reflect the rate of evolution at a particular time. Evolutionary activity is also claimed to be a straightforward method for studying evolvability [19]. The argument is that, since a system with high evolvability can create highly adaptive variation, the quantification of evolvability can be determined from different levels of extent and intensity of evolutionary activity.

In Biology, the rate of evolution comes in a few different flavors, depending on the objects being examined, such as gene sequences, proteins, and tissues, etc. Because of the infeasibility of defining fitness explicitly in natural systems, fitness is usually reflected by the likelihood that a relevant genetic change is selected. For instance, in molecular biology, the rate of evolution is usually measured by the rate that mutants are accepted and replace former alleles in genetic sequences. Biologists refer to this rate of evolution as *rate of genetic substitutions*.

We distinguish the rate of fitness progression and the rate of genetic substitutions to acknowledge the two aspects of rate of evolution. Fitness progression focuses on attaining the goal of the search, while rate of genetic substitutions concentrates on the dynamics of evolution and provides a different tool to study evolutionary processes. We are particularly interested in the rate of genetic substitutions measurement from molecular biology. Evolvability implies the potential to evolve. Rather than improving immediate fitness, it concentrates on significantly longer, or even open-ended

evolution, especially under changing environments. Evolutionary progress cannot be determined by how good population fitness is per se, but should be regarded as a “second-order” effect of fitness improvements. Therefore, we believe that the rate of evolution measured by the rate of genetic substations, should be a good complement looking beyond fitness.

3.2 The k_a/k_s ratio in Biology

In modern molecular evolution research, comparing the nucleotide (*A*, *C*, *G*, and *T*) sequences between related species is a method to study the process of evolution [111]. Biologists use the k_a/k_s ratio to measure the rate of divergence between two homologous protein-coding gene sequences from related species. This k_a/k_s ratio is defined as the ratio of the number of nonsynonymous (amino acid) substitutions per nonsynonymous site (k_a) to the number of synonymous substitutions per synonymous site (k_s).

In molecular biology, a *codon* comprises three ribo-nucleotides, and each codon determines one amino acid. A sequence of amino acids forms a protein, which produces the functional phenotype of an organism. Therefore, a codon is considered as a functional unit in evolution. A single nucleotide mutation at one out of the three sites on a codon will make this codon change to another one. Due to the redundancy of the genetic code, different codons may encode the same amino acid (e.g., codons *AAA* and *AAG* both code for amino acid lysine). Thus, a nucleotide mutation at a codon may be *synonymous*, which will not lead to amino acid substitution. If two different codons generated by a nucleotide change can produce different amino acids,

this nucleotide change is regarded as a *nonsynonymous* (amino acid) substitution.

To quantify how many nonsynonymous or synonymous sites are in a gene sequence, one has to characterize each site on a codon first. For a nucleotide location on a codon, all possible single-nucleotide mutations at this position are enumerated, and the fraction of those nonsynonymous (synonymous, resp.) mutations among all are calculated. Specifically, for a codon ε , if $f_\varepsilon(i)$ ($i = 1, 2, 3$) denotes the fraction of nonsynonymous single-nucleotide mutations among all possible single-nucleotide mutations at site i , the number of nonsynonymous sites at codon ε is $\sum_{i=1}^3 f_\varepsilon(i)$, and thus, the number of synonymous sites at codon ε is $3 - \sum_{i=1}^3 f_\varepsilon(i)$ [119].

A simple codon-based example for comparing two homologous gene sequences α and β is shown in Figure 3.1. The first pair of codons from both sequences are identical, coding for amino acid glutamine. The second pair of codons only have one difference at the last position, but they still code for the same amino acid lysine, while for the third pair, the nucleotides at the third position make codon 3 of gene sequence β code for amino acid aspartic, different from the amino acid lysine coded by codon 3 of gene sequence α .

The differences between two homologous gene sequences are counted by pairwise comparison of codons on these sequences. A nucleotide difference between these two sequences can be a nonsynonymous or synonymous substitution, depending on whether their contextual amino acid sequences are the same or not. Among these differences, the number of observed nonsynonymous nucleotide substitutions is denoted by M_n , and that of synonymous nucleotide substitutions is denoted by M_s . Further, the total number of possible nonsynonymous (synonymous, resp.) sites is calculated by summing up the numbers of possible nonsynonymous (synonymous, resp.) sites on

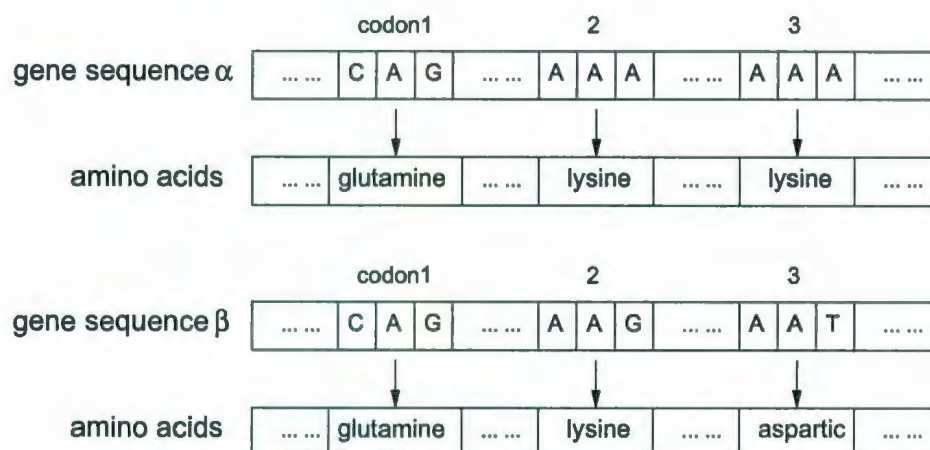


Figure 3.1: Codon-based sequence comparison.

each codon, denoted by N_a (N_s , resp.). Therefore, the nonsynonymous substitution rate $k_a = M_a/N_a$ is the number of observed nonsynonymous substitutions divided by the total number of possible substitutions. This is a metric of how much evolution has occurred in protein sequences normalized by all possible genetic variations between the two species. Similarly, rate $k_s = M_s/N_s$ is the number of observed synonymous changes divided by the total number of such changes that the sequence is capable of. This metric measures the “background” rate of “silent” genetic evolution without phenotypical change between the two species.

In reality, since the two homologous sequences have evolved for quite a long period of time, observing the number of differences between them underestimates the real number of changes. For instance, two changes from A to G then G to C may only be observed as a single change from A to C . Hence, multiple changes need to be considered and all possible pathways should be estimated when calculating M_a , M_s , N_a and N_s [187]. Although not adopted in our work, a few estimation methods have been proposed in the biological literature, and there are two major approaches among

them. The first one is based on the “approximate method” [119], which estimates all possible nucleotide substitutions and all possible pathways from one sequence to the other and assumes an equal rate for all types of nucleotide substitutions, while, in fact, different types of substitutions occur with varying likelihood. Therefore, many biases exist in gene sequence changes. The second approach is the “maximum likelihood method” [120], which uses some biased rates obtained from explicit models to estimate M_a , M_s , N_a and N_s .

After the two rates k_a and k_s have been estimated, their ratio can be used to quantify the rate of evolution. First, k_a measures the evolution of genetic changes producing variations in functional phenotype. The phenotypical variations have been selected according to the adaptation generated and have been fixed in the gene sequence. Second, k_s describes the rate of silent genetic changes being fixed in gene sequences, upon which selection pressure did not act. Therefore, k_a/k_s quantifies the rate of adaptive evolution by representing efficient evolutionary changes in relation to silent background evolutionary changes. This ratio also reflects the selection pressure on the evolution of organisms. In the case of $k_a/k_s > 1$, fixation of nonsynonymous substitutions is faster than that of synonymous substitutions, which means that *positive Darwinian selection* fixes amino acid changes faster than silent ones. While mostly one finds $k_a/k_s < 1$, the case where deleterious substitutions are eliminated by *purifying selection* (negative selection), and the rate of fixation of amino acid changes is reduced. If $k_a = k_s$, the fixation of these two types of changes are at the same rate. Measuring a large k_a/k_s ratio suggests that mostly observable changes has been generated and fixed at a high rate.

3.3 Applying k_a/k_s ratio in EC

Inspired by the k_a/k_s measurement on the rate of protein-coding gene sequence evolution in biology, we define the rate of evolution and propose a measurement for EC systems. An EC system with higher evolvability can generate efficient adaptation under selection pressure, so it has a good *potential* to improve fitness. Evidently, this capability or potential is less observable than fitness itself. Since rapid evolution is caused by generating adaptations at a high rate we can focus on the adaptive genetic changes underneath the phenotypical fitness to investigate the evolutionary progress. Here, we define the *rate of evolution* R_e as the rate of adaptive genetic changes being accepted into an EC system, similar to the rate of genetic substitution from Biology. Note that, non-synonymous substitutions are the only type of substitutions subject to selection but they are not necessarily adaptive. However, in EC, if a non-synonymous substitution can survive through selection, it should have improved the fitness of an individual, thus, it is adaptive.

Since selection acts at the phenotypical level, the adaptation of a genetic change to its environment can be determined by its acceptance into the population. Some changes that are able to improve the adaptation will be accepted, i.e., nonsynonymous substitutions, while other attempted deleterious changes will be eliminated. Some silent changes will be accepted as synonymous substitutions without experiencing selection pressure on phenotypical improvement. Dividing the rate of adaptive substitutions by the rate of synonymous substitutions can quantify the rate of adaptive evolution in an EC system. Therefore, if selection favors the innovated adaptive genetic changes at a high rate relative to the background rate, we say that this EC

system has a high rate of evolution.

However, as it is known that natural systems are much more complex than our EC systems, the approach that biologists use to measure the rate of sequence evolution with the k_a/k_s ratio should be transferred carefully into an EC system. Among others, the major differences between these two systems are:

- In biological measurement, two gene sequences from related species compared are considered having the same most recent ancestor, and this evolution mostly has proceeded over tens of thousands of generations. Biologists use a single value of k_a/k_s to measure the rate of divergence of two species from a most recent common ancestor. Whereas in an EC system, we calculate the rate of evolution for each well-observed generation such that we can obtain a time series of the k_a/k_s ratio.
- Since the two biological gene sequences have evolved for quite a long period of time, all those possible intermediate multi-step substitutions and all possible nucleotide substitutions at each site can only be estimated. Whereas in the generational evolution process of an EC system, all the changes and acceptance activities are monitored and can be traced accurately from the very beginning.
- The nucleotide sequences that biologists measure have a fixed length and each site can be precisely located; whereas in an EC population, e.g., in GP, syntax trees have varying sizes and undetermined sites.
- A genetic change in a biological system only refers to a random mutation on a gene sequence, whereas in an EC system, we usually consider both mutation and crossover as genetic changes.

Therefore, some aspects of the measurement definition can have corresponding metrics in an EC system, but others need to be replaced by EC-derived metrics.

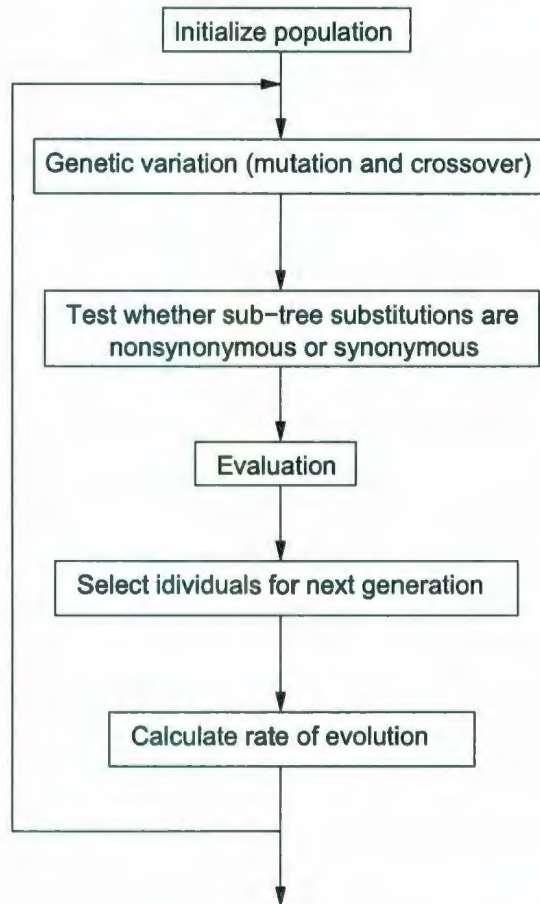


Figure 3.2: The flow chart with measuring rate of evolution in GP.

3.3.1 Case study: tree-based Genetic Programming

As a case study, first we utilize a tree-based GP system to implement this measurement [74, 75] since it is the basic representation for GP. We will test on Linear GP in later sections. For a GP tree in our case, genetic changes can be nonsynonymous as

in biological systems, which lead to representing different functions, or synonymous, which keep the encoded function unchanged. We calculate the number of substitutions and divide it by the “sites” for a GP system to obtain the two types of rates. Here, we measure the rate of evolution for a GP system in each generation. Specifically, before establishing a generation t , standard mutation and crossover, limited to subtree replacement, are applied to the individual trees in a GP population of generation $t - 1$. Tournament selection with truncation scheme is then performed on both the parents and offspring to form the next generation t . In such an iteration, we define the rate of evolution $R_e(t)$ of generation t by observing the individual genetic changes and their acceptance into the population. Figure 3.2 shows the implementation of this measurement in a GP system.

It is well known that changes to a GP tree may be silent due to the existence of neutral *intron* codes [16]. That is, syntactic changes to a tree may or may not lead to functional changes. Therefore, after mutation or crossover of the trees, these subtree replacements are either nonsynonymous or synonymous. For each individual tree i , if a change is silent, the value of nonsynonymous change $m_a^i(t)$ is set to 0 and the value of synonymous change $m_s^i(t)$ is set to 1. In contrast, if a change leads to functional differences, $m_a^i(t)$ is 1 and $m_s^i(t)$ is 0. If tree i is not modified from generation $t - 1$ to generation t , both $m_a^i(t)$ and $m_s^i(t)$ remain 0. After the truncation tournament selection chooses new individuals from both the parents and offspring, a new generation t is established. As a result, the total number of nonsynonymous substitutions $M_a(t)$ and synonymous substitutions $M_s(t)$ for the entire population of

generation t can be calculated as

$$M_a(t) = \sum_{i=1}^S m_a^i(t) , \quad M_s(t) = \sum_{i=1}^S m_s^i(t) , \quad (3.1)$$

where S is the population size. Note that, $M_a(t)$ and $M_s(t)$ only count those genetic changes accepted into the population that have survived through the selection.

As we discussed in the biological k_a/k_s ratio definition (Section 3.2), the numbers of nonsynonymous sites and synonymous sites represent the *potential* of the sequence to produce nonsynonymous or synonymous changes, and are used to “normalize” the numbers of substitutions. Here, we adopt a *sensitivity* notion to describe the potential of a GP tree to change its semantic meaning in the event of a subtree replacement. Trees have varying sensitivities against subtree replacements, an observation made by Langdon and Banzhaf [99] in research on repeated patterns in tree-based GP systems. We keep a record of all changes to a tree from the beginning of evolution including all attempted subtree replacements, such that the accumulated fraction of these changes being nonsynonymous or synonymous can be regarded as the nonsynonymous sensitivity and synonymous sensitivity of this tree. Specifically, for an individual tree i after initialization, we use $c_a^i(t)$ and $c_s^i(t)$ to denote the accumulated numbers of nonsynonymous and synonymous changes of generation t , respectively, obtained by summing up all the previously recorded changes that have happened to this tree,

$$c_a^i(t) = c_a^i(t-1) + m_a^i(t) , \quad c_s^i(t) = c_s^i(t-1) + m_s^i(t) , \quad (3.2)$$

with

$$c_a^i(0) = c_s^i(0) = 0 . \quad (3.3)$$

Therefore, the nonsynonymous and synonymous sensitivities of tree i of generation t can be obtained as follows from the fraction of each type of changes, and these metrics

indicate the degree of tree i being changed nonsynonymously or synonymously,

$$n_a^i(t) = \frac{c_a^i(t)}{c_a^i(t) + c_s^i(t)} , \quad n_s^i(t) = \frac{c_s^i(t)}{c_a^i(t) + c_s^i(t)} . \quad (3.4)$$

We add up the sensitivities of all individuals in the population to obtain the total nonsynonymous and synonymous sensitivities as the “sites” of the current generation,

$$N_a(t) = \sum_{i=1}^S n_a^i(t) , \quad N_s(t) = \sum_{i=1}^S n_s^i(t) . \quad (3.5)$$

Last, we define the nonsynonymous and the synonymous substitution rates k_a and k_s of generation t as

$$k_a(t) = \frac{M_a(t)}{N_a(t)} , \quad k_s(t) = \frac{M_s(t)}{N_s(t)} . \quad (3.6)$$

The rate $k_a(t)$ measures the rate of generating nonsynonymous adaptive changes. The rate $k_s(t)$ describes the rate of producing neutral changes in an evolutionary process. Without changes at the functional level, these neutral changes will not experience pressure in evolution. Thus, $k_s(t)$ practically provides “clock ticks” for the acceptance of genetic changes in the GP system. Since $k_a(t)$ measures the rate of accepted effective changes, the ratio $k_a(t)/k_s(t)$ represents the “evolutionary distance” in relation to the “evolutionary time”, therefore, the rate of effective adaptation of generation t . Thus, we propose the rate of evolution R_e in the GP tree population of generation t to be

$$R_e(t) = \frac{k_a(t)}{k_s(t)} . \quad (3.7)$$

3.3.2 Simulation with symbolic regression

We calculate R_e using GP to solve a benchmark quintic polynomial symbolic regression problem $x^5 - 2x^3 + x$ defined by Koza [92]. Each individual in this GP population

is a syntax tree initialized by the method *ramped half-and-half* with maximum depth 6. Candidate functions are evolved toward a target function $f(x) = x^5 - 2x^3 + x$ within interval $[-1, 1]$ by matching a set of sample points. The sample set has 50 real numbers uniformly distributed in $[-1, 1]$. The absolute difference between output and the target $f(x)$ value is the error, and the fitness function is defined as the average error over all 50 samples. The terminal set includes variable x and random ephemeral constants generated from 2001 numbers equally distributed in $[-1, 1]$ with granularity of 0.001. The four arithmetic operators: $+$, $-$, \times , and protective \div are used as the function set. We apply random mutation and crossover with probabilities 0.1 and 0.9, respectively, and the maximum mutation subtree depth is 4. Parent individuals and offspring after genetic changes compete through truncation selection with tournament size 4. This GP system has a population size of 4000 evolved for a maximum of 50 generations. A set of 20 cases are used as inputs to a GP tree before and after mutation or crossover, to test whether a subtree replacement is nonsynonymous or synonymous. If all 20 cases produce the same output, subtree replacement applied to this tree is regarded synonymous; otherwise, this tree is considered to have undergone a nonsynonymous change.

3.3.2.1 Single evolution process

First, we measure R_e in a single GP run to observe how it changes during different stages of an evolutionary process.

We chose a successful run that has reached the target function within the termination criterion of 50 generations. Figure 3.3 (a) shows the best and average fitness of the entire population for this run. Since the fitness function is calculated as the

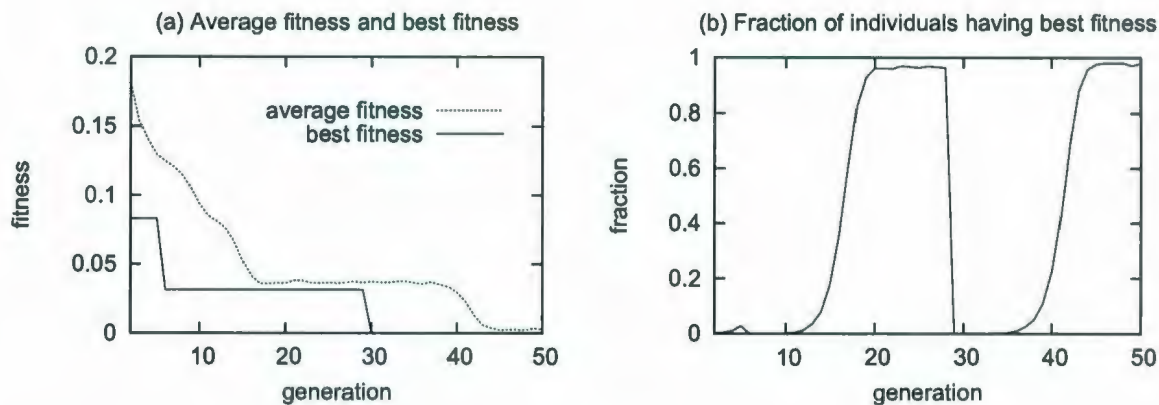


Figure 3.3: Fitness and diversity development in a single GP evolution.

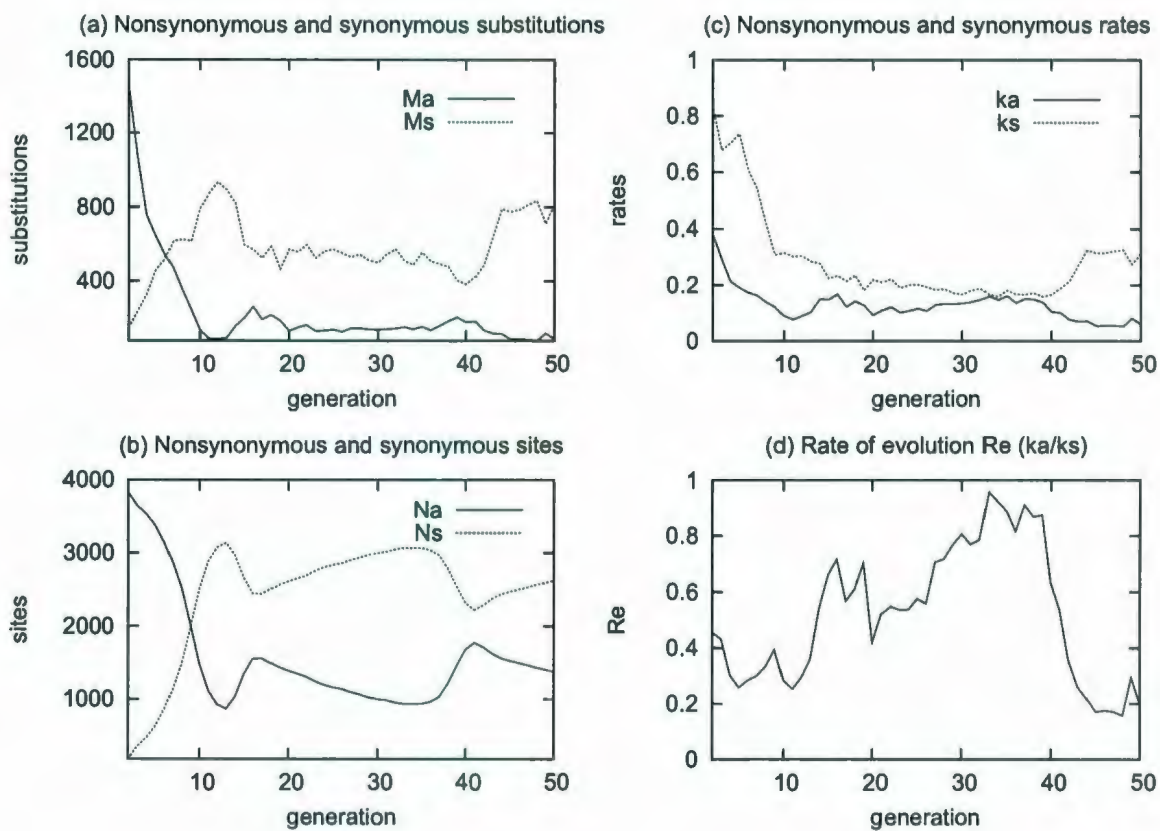


Figure 3.4: The k_a/k_s measurement in a GP evolution.

difference between the evolving function and the target function, those two types of fitness-based metrics approach zero as evolution proceeds. These fitness-based curves are what we usually investigate when evaluating the performance of a GP system. In addition, we calculate the fraction of the individuals that have the same best fitness. These individual trees may not have the same structure, but they encode the same function. This fraction is depicted in Figure 3.3 (b) showing that, from about generation 12 to 28, an individual with fitness 0.03 propagates in the population, while later between generation 33 to 43, the individual representing the target function approaches predominance in the population. About 98% of the individual trees represent the target function during the end phase of this evolutionary process, because a relatively low truncation tournament selection size of 4 allows some noise at the end of evolution.

The number of substitutions M_a and M_s , the number of sites N_a and N_s , the substitution rates k_a and k_s , and the rate of evolution R_e are shown in Figure 3.4. The number of nonsynonymous changes M_a is large at the start, but decreases dramatically until the dominant individual is produced. Then it increases slightly during the propagation of the predominant individual. The number of synonymous changes M_s increases from the start of evolution, then it begins to decrease during the two propagation processes of dominant individuals, and finally it increases after 98% individuals are representing the target function. This is because after most trees begin to represent the same function, some patterns are growing in numbers in these trees to maintain the same genotype. The number of nonsynonymous sites N_a , which indicates the sensitivity of individual trees towards nonsynonymous changes, generally decreases because introns generated by genetic operations affect GP trees decreasingly

sensitive to nonsynonymous subtree replacements. This can also be seen in Figure 3.3 (c) that trees are growing larger during evolution. And this sensitivity only rebounds slightly during the two dominant tree propagation processes. As the complement number of sites in the entire population, N_s increases in general. The nonsynonymous substitution rate $k_a = M_a/N_a$ reflects the rate of adaptive evolution by amplifying the adaptive innovations divided by decreasing tree sensitivities, and $k_s = M_s/N_s$ provides the rate of the silent changes being generated and accepted. The rate of evolution R_e , which is calculated as the ratio k_a/k_s in Figure 3.4 (d), shows the rate of innovating adaptation in this GP evolution. The evolution process under fitness development is well reflected by R_e .

In this GP evolutionary process, plotting the rate of evolution R_e shows the adaptive substitutions and indicates the rate at which the evolutionary search proceeds. The value of R_e is below 1.0 which accords well with the situation in natural evolution, meaning that most attempted random genetic changes are deleterious and selection acts mainly to purify these harmful random changes.

3.3.2.2 Comparisons with varying parameter setting

Here, we compare R_e in different configuration scenarios by varying such parameters as selection size, population size, mutation rate, and crossover rate, to study their effects on the rate of evolution and to verify the effectiveness of our approach. In each set of experiments, only the investigated parameter is changed and all others are held constant. The average fitness, k_a , k_s and R_e are plotted with the average values of 50 successful runs. The method *exponentially weighted moving average* is used here to smooth the curves (smoothing factor 0.1).

Tournament selection size

We increase tournament selection size from 4 to 6 and to 8 (Figure 3.5). It is generally accepted that a larger tournament selection size generates greater survival pressure [24], and thus can maintain a better fitness in the population. It can be seen that the population under tournament selection size 8 has the best average fitness. However, due to a higher selection pressure, fewer innovative individuals are accepted, so the population with tournament size 8 has the lowest nonsynonymous substitution rate k_a . In contrast, relatively more silent changes are accepted with a larger tournament selection size. This also concurs with a recent prediction by Luke and Panait [108] that bloat of neutral code in GP is caused by the pressure of improving fitness. Therefore, the rate of evolution R_e decreases as the tournament size increases. These results show that higher selection pressure slows down the rate of accepting genetic variations.

Population size

We test the GP system with different population sizes 200, 2,000 and 20,000 (Figure 3.6). Observe that a larger population is better at searching and maintaining the average fitness. All three nonsynonymous substitution rates k_a with different population sizes are quite close, which indicates that, although larger populations offer a larger amount of adaptive individuals to be generated and accepted, their rates in this static symbolic regression problem are nearly the same as smaller populations. Further, a larger population accepts synonymous genetic changes at a slower rate, which is an expected result of a slower propagating speed of dominant individuals. It

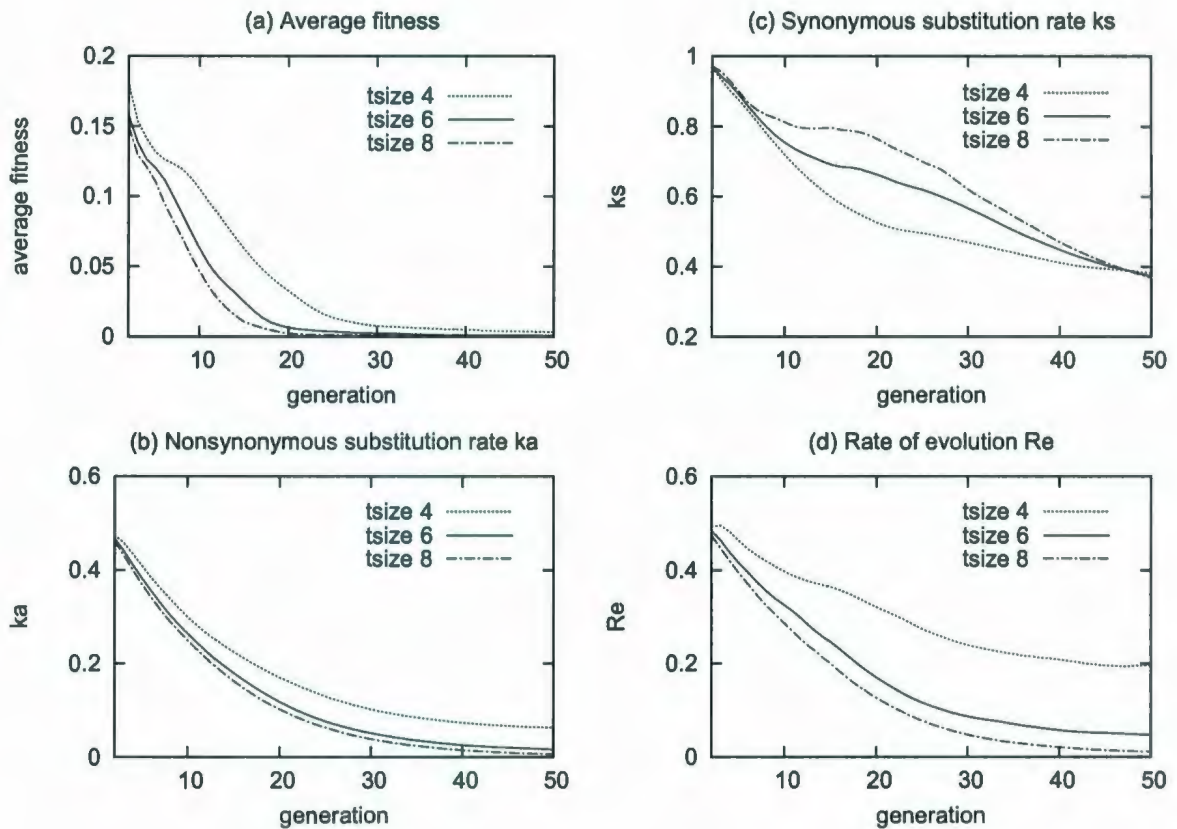


Figure 3.5: Evolution with different tournament selection sizes.

can be observed that a larger population has a slightly higher R_e at the early stage of the search process but slows down when the target individual becomes dominant in the population. These differences are quite small, however, for this static optimization problem. So we believe that, although a larger population offers more chances of innovating adaptation, under the same environment and selection pressure, a larger population does not have a real constant advantage in improving the rate of evolution. It can be seen further that the population with size 200 has the most drastically changing rates, accepting genetic changes at a fairly high rate even around generation 50 (see also the average fitness chart).

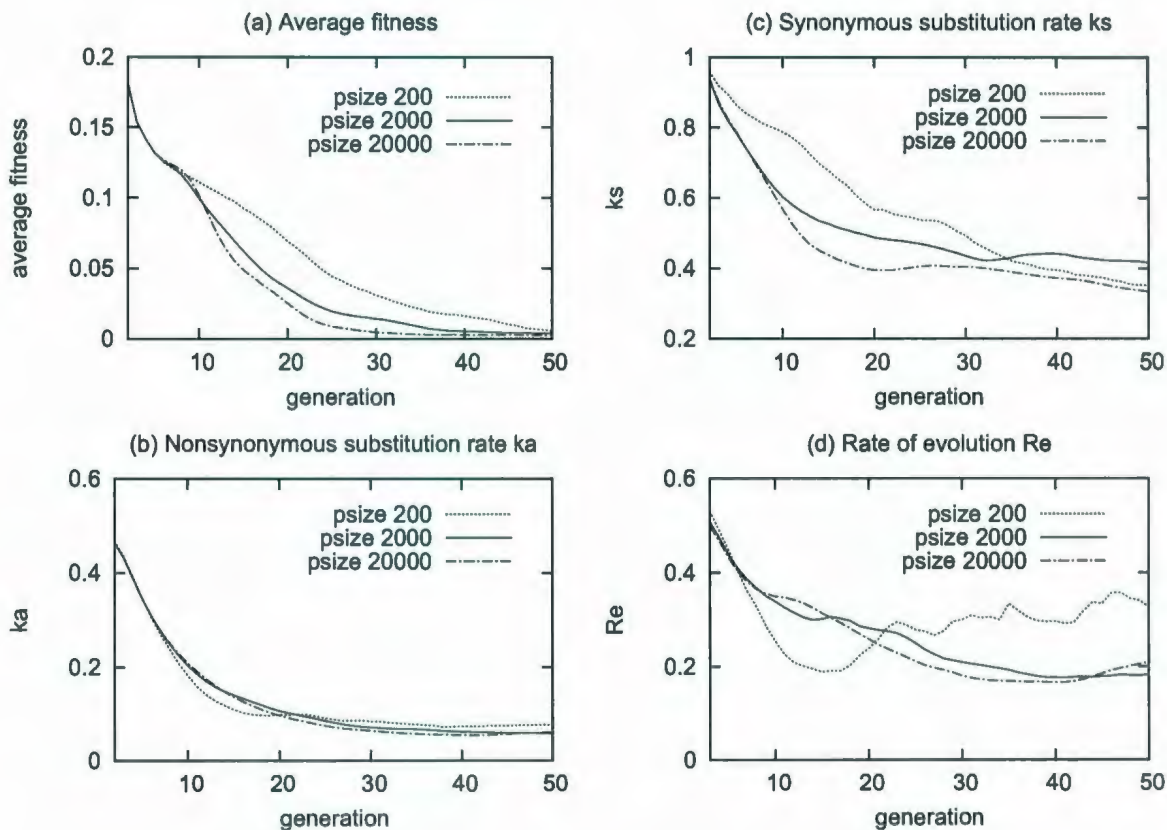


Figure 3.6: Evolution with different population sizes.

Mutation rate

The mutation rate is set to 0.3, 0.6, and 0.9 when the crossover rate is fixed to 0.1 (Figure 3.7). In our simulations, we only collect successful runs which can reach the target function within 50 generations. A population with a higher mutation rate is more likely to succeed. We observed that the percentages of successful runs with mutation rates 0.3, 0.6, and 0.9 are 16%, 22%, and 30%. However, despite different success likelihoods, various mutation rates do not show considerable differences in the rate of improving the average fitness solving this problem. In our rate of evolution measurement, it can be observed that, a higher mutation rate results in a higher

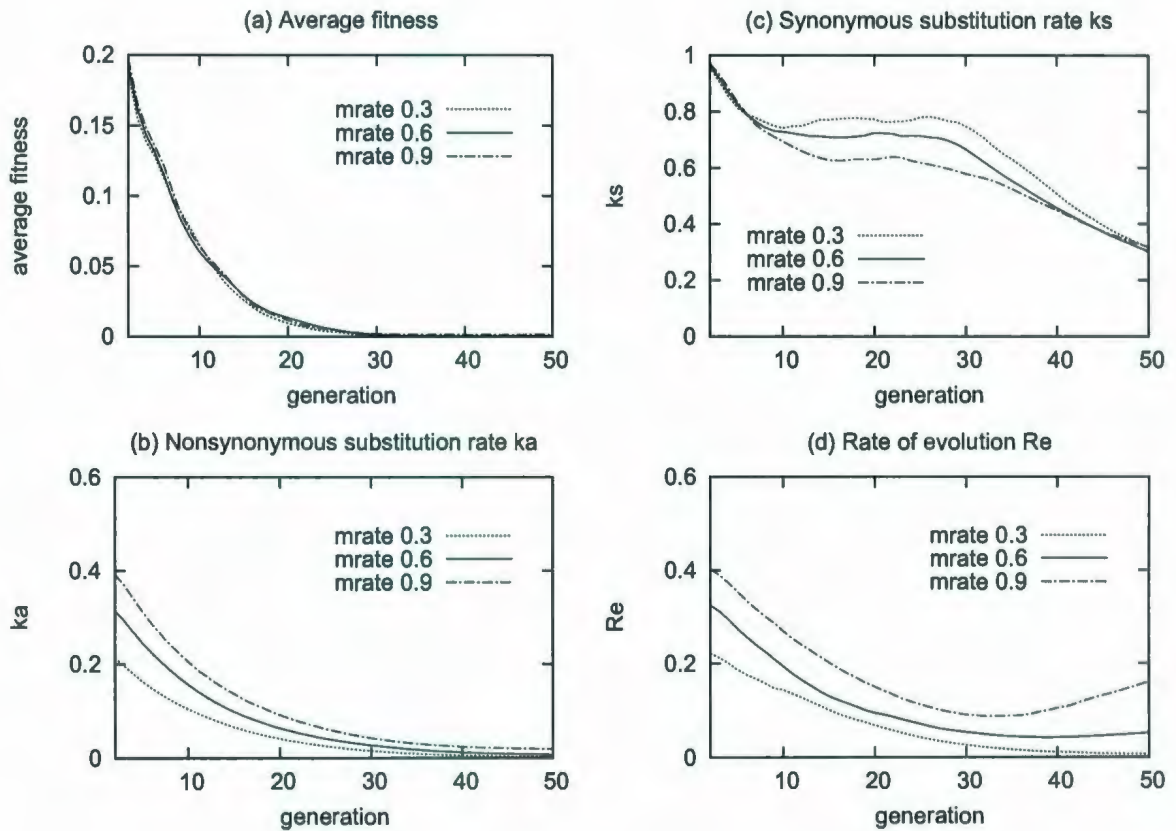


Figure 3.7: Evolution with different mutation rates.

nonsynonymous substitution rate k_a and a lower synonymous rate k_s , and thus, a higher evolution rate R_e . These results show that a higher mutation rate can accelerate evolution but also brings in more noise at the end of evolution (Figure 3.7 (d)). Moreover, this simulation supports a general tendency of mutation to maintain high population diversity.

Crossover rate

In this set of simulations, we fix the mutation rate at 0.1 and increase the crossover rate from 0.3 to 0.6, and to 0.9. In Figure 3.8, similarly to varying mutation rates, we

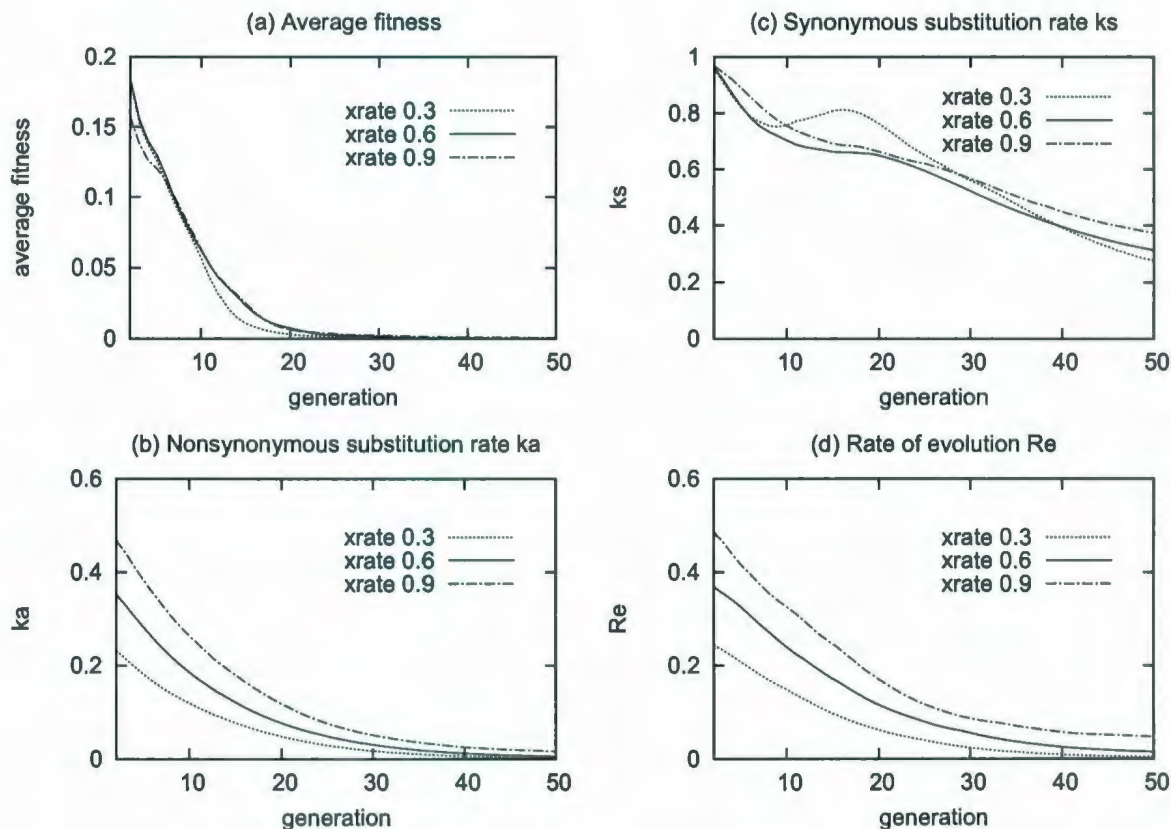


Figure 3.8: Evolution with different crossover rates.

can see that investigating fitness development is not sufficient for drawing conclusions on the effectiveness of crossover with regard to rate of evolution. In our measurement, it is observed that a larger crossover rate provides more adaptive genetic changes, i.e., a greater k_a , and consequently a higher rate of evolution R_e . However, the differences between mutation and crossover operations are their effects on synonymous substitution rate k_s . That is, increasing the crossover rate can result in a higher synonymous rate, which implies that crossover contributes more to neutral evolution than mutation.

3.4 Discussion

In this chapter, we introduced the equivalent of the biological measurement of the nonsynonymous to synonymous substitution ratio k_a/k_s into a GP system. The experimental applications show the ability of this measurement to capture the rate of generating efficient genetic variations in EC. Therefore, we believe that defining and measuring the rate of evolution as the rate of adaptation being generated and accepted, i.e., rate of genetic substitutions, can be very effective to capture an evolution process. By looking beyond fitness, this measurement provides observation into the level of evolution dynamics.

Further, some observations show that in the truncation selection scheme tournament size, mutation rate, and crossover rate are directly related to the rate of evolution, while population size has an indirect relation. These results suggest a non-monotonic relationship between population size and rate of evolution. This finding motivates the next step investigation, the role of population size in rate of evolution.

Chapter 4

Role of Population Size in Rate of Evolution

The search process in EC systems is a simultaneous process of exploration in parallel and exploitation in depth. Population size is a key factor to maintain population diversity in this process, and is thus critical for the performance of an EC method. Recently, population size control has attracted increasing interest in the literature [105].

Population size control is non-trivial and challenging because it is often problem-specific and the interaction among various EC parameters is not completely clear yet. In general, the literature on population size control has two main foci: i) initializing a proper population size a priori, and ii) adjusting population size during evolution. We focus on the latter.

Population size adjustment is motivated by the observation that the required population size changes during different stages of evolution [10]. Such an adjustment is usually directed by a feedback loop. This feedback has been implemented through

the controlled persistence of individuals or through the measurement of fitness progression, both of which are able to reflect the process of evolution to some extent.

In Biology, particularly in the study on *population genetics*, population size has been intensely studied regarding its role in the rate of evolution [131, 132]. It has been realized that the effect of population size on evolution acceleration is conditioned on the nature of selection at a particular moment rather than on a monotonic relationship. Typically, under *positive selection*, i.e., selection mostly accepting adaptive phenotypes, a large population is favorable for rapid evolution. In contrast, under *negative selection*, i.e., selection mostly eliminating deleterious phenotypes, a small population evolves faster. These two selection conditions can be reflected by the rate of genetic substitutions.

Although this perspective is still under debate in the biological community, it is intriguing to study this relationship in EC systems. Thus, we investigate the interplay between population size and rate of evolution in an EC model and see how this originally biological notion translates to artificial systems [77].

4.1 Background and motivation

In this section, we first briefly review studies on population size control in EC. Then, we discuss the relation between population size and the rate of evolution from both an EC and a biological point of view.

4.1.1 Population size control

Research on population size control in EC originated from *Genetic Algorithms* (GAs). A number of theoretical contributions on analyzing population size initialization have been published based on Goldberg's seminal "components decomposition approach" and the notion of *building blocks* [59, 61]. The essence of these works is that population size should be initialized according to the "complexity" of a specific problem. That is, for a more difficult problem, more diversity of a population is required, and thus a larger population size should be initialized.

Recently, it was realized that even for a given problem instance the required population size can vary during the process of evolution. Therefore, besides a good initial population size, some empirical methods on adjusting population size dynamically have been proposed. Arabas et al. [8] propose the *Genetic Algorithm with Variable Population Size* (GAVaPS) by regulating the *age* and *lifetime* of each individual. Population size fluctuates as a result of removing over-aged individuals and reproducing new ones. Back et al. [10] extend this lifetime notion in their *Adaptive Population size Genetic Algorithm* (APGA) to steady-state GAs. Fernandes and Rosa [46] propose the *Self-Regulated Population size Evolutionary Algorithm* (SRP-EA) to enhance APGA using a diversity-driven reproduction process. Alternatively, Harik and Lobo [67] introduce *parameter-less GA*, where several populations with different sizes evolve in parallel, starting with small population sizes. By inspecting the average fitness of these populations, less fit undersized populations are replaced by larger ones. Eiben et al. [42] suggest to use the pace of fitness improvements as the signal to control population size in *Population Resizing on Fitness Improvement GA* (PRoFIGA).

In GP, determination of an ideal population size is of even greater significance. As with the GA, population size in GP is relevant to its capabilities in finding the target and to its computational efficiency. In particular, it is related to the phenomenon of *bloat*, i.e., the increasing size of program code in GP evolution without a corresponding improvement in fitness. Poli et al. [139] establish that smaller populations bloat at a slower rate than larger ones. Downing [37] investigates population size in relation to evolvability in GP. Thus, adjusting population size dynamically benefits GP in various ways. A theoretical analysis on population size in GP based on building blocks is conducted by Sastry et al. [153]. The empirical population size adjustment schemes for GAs can also be applied to GP. Moreover, some GP-specific techniques have been employed as well. For example, Wedge and Kell [179] propose the *Genotype-Fitness Correlation* as a landscape metric to predict ideal population sizes in different systems. Tomassini et al. [170] design a dynamic population size GP using fitness progression as a signal to delete over-sized and worse-fit individuals or to insert mutated best-fit individuals with certain criteria.

4.1.2 Population size and rate of evolution

In EC, the “goal” of evolution is very specific: to find the fittest solution to a given problem. In this sense, as long as an EC population is able to find solutions, a small size is favored because of a small overhead. Thus, computer scientists have been seeking intelligent population size control schemes to strike a balance between exploration and exploitation during the search process.

In Biology, population geneticists have been identifying the effect of population

size on the rate of molecular evolution, i.e., the rate of genetic substitutions. The *Nearly Neutral Theory* of molecular evolution by Ohta [131, 132] is regarded as one of the most important principles for modern molecular evolution research. This theory defines both slightly deleterious and slightly advantageous mutations as *nearly neutral mutations*. It extends an earlier insight of Fisher [47] that the probability of a mutant being selected will be low if the outcome of this mutation on phenotypes is far-reaching. The theory predicts that most substitutions are neutral or nearly neutral in molecular evolution. These nearly neutral mutations would be able to generate adaptation at a later time under certain genetic or environmental changes. Thus, they play an important role for providing variation potential.

In this theory, population size can influence the rate of molecular evolution by its effects on the chance of accepting a nearly neutral genetic change through statistical laws. That is, the chance of a random mutant being fixed by selection is less within a larger population. When the majority of mutations are deleterious, a smaller population can evolve faster because more nearly neutral changes are introduced into the population. In contrast, when mutations are mostly advantageous, evolution is faster in a larger population. When most mutations are neutral, the rate of evolution is nearly independent of the population size.

These predictions have been extensively tested and discussed in the biological community. Below are some examples where both increasing and decreasing population size may accelerate evolution, depending on the link to environments. Gillespie [56] examines the relation between population size and the rate of genetic substitutions via computer simulation of several well-known biological models. While verifying such a relation, he suggests the relation can sometimes be blurred by the extreme

complexity of natural systems. With population size fluctuation being one of these complicating factors, he further emphasizes the necessity of studying such fluctuation in population genetics. Woolfit and Bromham [183] compare genetic sequences between island endemic species and closely related mainland species. This is an example where decreasing population size can accelerate evolution. In a study on the recent rapid molecular evolution in human genomes, Hawks et al. [70] hypothesize that the current dramatically growing human population may be the major driving force of new adaptive evolution. They indicate that a growing population size can provide the potential for rapid adaptive innovations if a population is highly adapted to the current environment.

4.2 Adjusting population size during evolution

We propose to apply the ideas from population genetics to a GP system. It is generally assumed that the fitness of new offspring generated by mutation or crossover in each generation approximately follows a Gaussian distribution. According to the *Central Limit Theorem*, the average fitness among a larger population has a smaller variance [131] (Figures 4.1 and 4.2).

We use the *selection favoring degree* S_f to denote the degree of new offspring being favored by selection. A positive value of S_f of an offspring implies that it is likely to be accepted, and a negative value of S_f means that it will most likely be rejected by selection. Further, if the majority of offspring have positive S_f , selection is referred to as positive (Figure 4.1). In contrast, the selection is negative when $S_f < 0$ for most offspring (Figure 4.2). From the figures, we observe that, under positive selection,

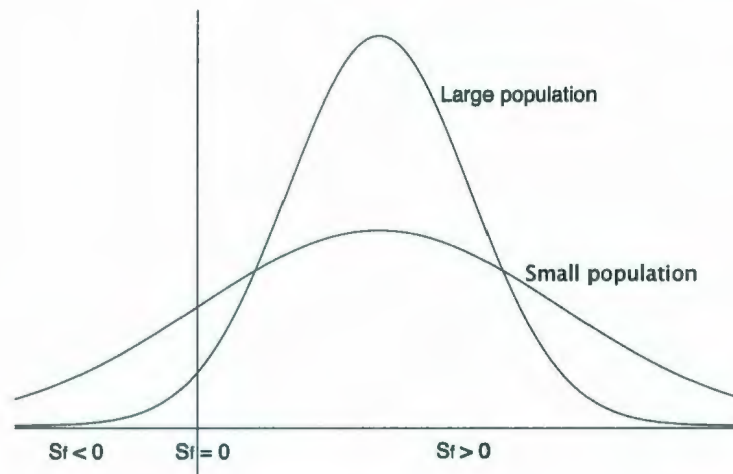


Figure 4.1: Positive selection in different sized populations.

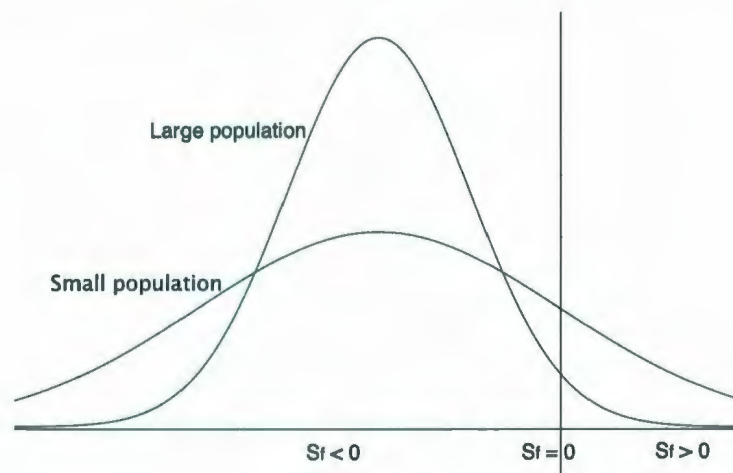


Figure 4.2: Negative selection in different sized populations.

increasing population size can accelerate the rate of genetic substitutions, while under negative selection decreasing population size can allow more genetic substitutions.

Selection acting positively or negatively may vary during different stages of an evolutionary process in GP, and the rate of genetic substitutions reflects this varying selection pressure. Therefore, adjusting population size according to the rate of

genetic substitutions is expected to compensate for the selection pressure. Thus, evolution can be guided away from stagnation and, further, can achieve better fitness progression. A slightly revised measurement k_a/k_s ratio for the rate of genetic substitutions is described briefly next, followed by our proposed population size adjustment approach.

4.2.1 Adjustment indicator: rate of evolution

We slightly revise the k_a/k_s ratio proposed in the previous chapter, to measure the rate of genetic substitutions. We aim to simplify the calculation of this ratio since the original method is considered time-consuming.

From one generation to the next, N_a denotes the number of all *attempted* nonsynonymous genetic changes, and M_a counts the number of *accepted* nonsynonymous genetic changes. A *sampled semantic test set* different from the training set is fed to an individual before and after a genetic change to test whether this change is nonsynonymous or synonymous. For instance, if a parent and its offspring have the same output for all sampled semantic test cases, the genetic change generating this offspring from the parent is regarded as synonymous. Otherwise, the change is considered nonsynonymous.

Thus, $k_a = M_a/N_a$ measures the rate of accepting nonsynonymous genetic changes. The synonymous substitution rate k_s can be defined similarly by dividing the number of accepted synonymous genetic changes M_s by the number of attempted synonymous genetic changes N_s . The ratio k_a/k_s measures the rate of adaptive (since they are accepted) genetic substitutions relative to a background silent genetic substitution

rate. The case $k_a/k_s = 1$ corresponds to the situation where nonsynonymous genetic changes are selected at the same rate as neutral changes. When $k_a/k_s > 1$, selection is positive because a larger portion of nonsynonymous changes are favored by selection. In contrast, negative selection is reflected by the case $k_a/k_s < 1$.

4.2.2 Adaptive population size approach

Next, an adaptive population size scheme is proposed using the k_a/k_s ratio defined above. We adopt truncation selection such that population size adjustment can be achieved easily without duplication or generating random individuals. Typically, at generation t , the current population produces an offspring population of the same size via genetic variations including crossover and mutation. Parents and offspring will compete through tournament selection to yield the next generation, and the population size $P_{\text{size}}(t + 1)$ will be adjusted according to the currently observed rate of genetic substitutions $(k_a/k_s)(t)$. Thus, the adaptive population size is regulated in each generation in an attempt to maintain a stable ratio of genetic substitutions as follows:

- If $(k_a/k_s)(t) > 1$ (positive selection), we increase the population size proportional to the changes of the rate of genetic substitutions such that,

$$P_{\text{size}}(t + 1) = P_{\text{size}}(t) \times (1 + |(k_a/k_s)(t) - (k_a/k_s)(t - 1)|).$$

- If $(k_a/k_s)(t) = 1$ (neutral selection), we keep the same population size,

$$P_{\text{size}}(t + 1) = P_{\text{size}}(t).$$

- If $(k_a/k_s)(t) < 1$ (negative selection), when $(k_a/k_s)(t)$ is increasing, we increase the population size to suppress further deleterious genetic substitutions, and when $(k_a/k_s)(t)$ is decreasing, we decrease the population size to encourage more genetic substitutions. That is,

$$P_{\text{size}}(t+1) = P_{\text{size}}(t) \times (1 + (k_a/k_s)(t) - (k_a/k_s)(t-1)).$$

Note that we do not limit the population size by an upper bound. However, a lower bound on population size will be established in applications.

4.3 Simulation with tree-based GP

We expect that dynamic adjustment of population size according to the measured k_a/k_s ratio can maintain a fairly stable rate of genetic substitutions. Since evolution seems to be better guided this way, the performance of a GP system in fitness progression should improve as well. This is verified through simulations and comparisons to fixed-size populations. A tree structure is adopted to encode GP individuals here. The test benchmark will be introduced next, followed by our discussion of experimental results.

4.3.1 Test suite: Mackey-Glass time series

We use the Mackey-Glass chaotic time series prediction as our benchmark problem. The Mackey-Glass time series prediction is a difficult modeling problem in machine learning and in GP [129]. It predicts future values of a time series based on its historical values. GP is trained using these historical data. The series is generated

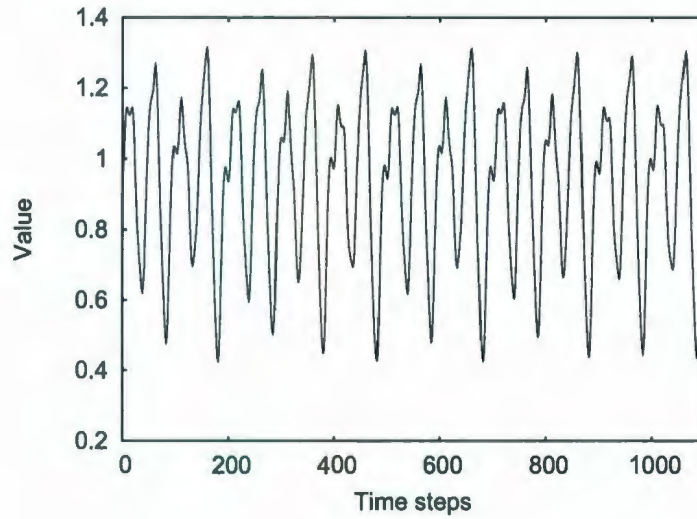


Figure 4.3: Mackey-Glass time series.

using the following recursive function [184],

$$x_{t+1} = x_t - b \times x_t + \frac{a \times x_{t-\tau}}{1 + (x_{t-\tau})^{10}},$$

where $x_0 = 1$, and the parameters are set to

$$a = 0.2, b = 0.1, \tau = 17.$$

Figure 4.3 depicts a plot of this function. We use the first 1,001 points as the training set. This problem is considered a difficult one because it does not have a closed-form solution. Thus, it will take GP a long time to converge.

Empirically, a population size between 500 and 1,000 is suitable for this type of problem. Here we conduct experiments in three scenarios. Two of the scenarios have fixed-size populations of 500 and 1,000, the third has an adaptive population size (APS) using our dynamic adjustment approach. It starts with an initial population size of 1,000 and a lower limit of 300. The GP configuration is as shown in Table 4.1.

Table 4.1: Tree-Based GP configuration for the Mackey-Glass time series.

Population size	500/1,000/APS(Adaptive Population Size)
Tree initialization	Ramped-Half-and-Half with limit 6
Function set	+, -, ×, and protective /
Terminal variable set	x_1, x_2, \dots , and x_{17} , variable x_i denotes the previous point i time steps ago
Terminal constant set	Random ephemeral numbers equally distributed in $[-1, 1]$ with granularity 0.01
Crossover rate	0.9
Mutation rate	0.1
Maximum mutation subtree depth	4
Crossover and mutation method	Subtree replacement
Maximum tree depth	100
Training set	Points from 0 to 1,000 time steps
Fitness function	Root Mean Square (RMS) error
Selection	Tournament with size 4
Sampled semantic test set	20 cases such that $x_i^j = (i + j - 2) \times 0.04$ ($1 \leq i \leq 17, 1 \leq j \leq 20$), $0 \leq x_i^j \leq 1.4$
Maximum number of evaluations	100,000

Note that we adopt the number of function evaluations as a control metric although it operates in a generational mode. This allows a fair comparison among different scenarios.

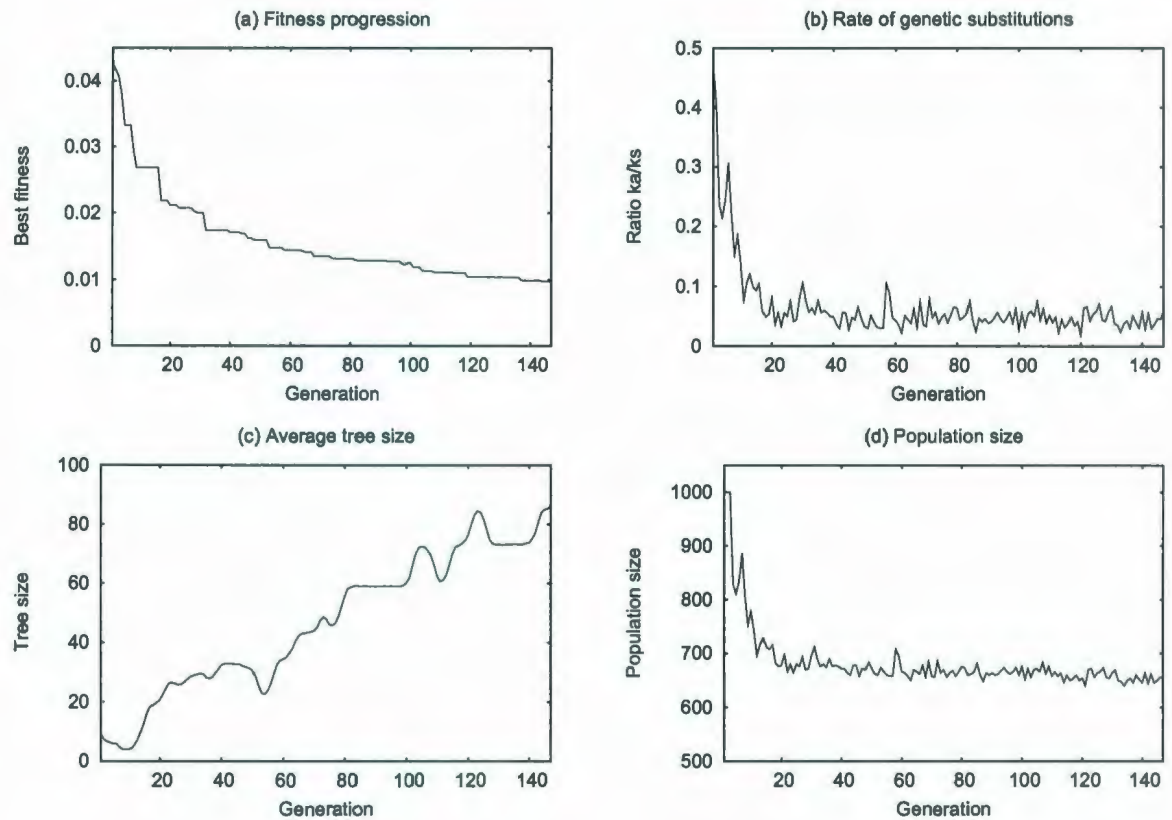


Figure 4.4: An example GP run with adaptive population size.

4.3.2 Results

We have run GP 200 times for each scenario. Before we present statistical results, we look into the details of a “typical” execution of the APS scenario (Figure 4.4). This particular population evolves for 147 generations before it reaches the 100,000 function evaluation number limit. In the figure, we plot (a) best fitness, (b) k_a/k_s ratio, (c) average tree size, and (d) population size over generations.

We observe that the k_a/k_s ratio stays well under 1, which implies that selection is negative over time. This concurs with the general understanding that attempted random genetic changes are mostly deleterious and with the property of the k_a/k_s ratio

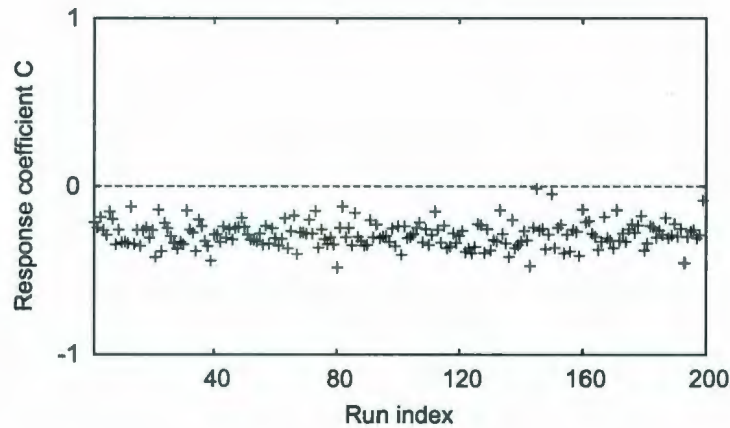


Figure 4.5: Correlations between population size and k_a/k_s ratio.

in Biology [187]. The population size drops from an initial 1,000 to approximately 700 after 20 generations, and stabilizes at 650-700 afterwards. Also notice that, as evolution progresses, the best fitness improves but at a slower rate, and average tree size increases, which is expected for tree GP. Normally, bloat would slow down the rate of genetic substitutions due to the introduction of redundant substructures. However, this is successfully alleviated by adjusting the population size to stimulate evolution so that there is a steady k_a/k_s ratio. This is verified by our next study of the interaction between the k_a/k_s ratio and population size.

In Figure 4.5, we depict the response of the k_a/k_s ratio change to population size adjustment, derived from the data recorded from 200 runs of the APS scheme. Using the recorded population size and the k_a/k_s ratio of each run, we quantify the correlation between the way they change over generations using a sequence of 1's and -1's. For a generation compared to the previous, if both population size and the k_a/k_s ratio increase, or if both population size and the k_a/k_s ratio decrease, we have a 1. Otherwise, we have a -1. Therefore, the number of 1's in the produced sequence

records the number of occurrences where the change of the k_a/k_s ratio positively correlates to that of the population size; while -1 's indicate negative correlation.

We define the *response coefficient* C as $C = (2n - l)/l$, where n is the number of 1's in the sequence and l is the sequence length. Thus, if a run has $C = 0$, its k_a/k_s ratio is independent of the change of population size. Alternatively, a positive value of C indicates a positive correlation between the changes of the k_a/k_s ratio and that of the population size. On the other hand, a negative value of C suggests a negative correlation. The figure presents the coefficients for all the 200 simulation runs. Clearly, they are all well below the level of 0. This is indeed our intention of dynamically adjusting population size to stabilize the rate of genetic substitution as stated in Section 4.2.2.

Our next observation is that fitness progression can also be accelerated by our population size adjustment scheme. Here, we adopt the three most commonly used metrics to measure the performance of an EC model. They are mean best fitness, success rate, and average number of evaluations to a solution.

Table 4.2: Best fitness ($\times 10^{-3}$) comparison with different sized populations.

	Mean	Standard Deviation	Median	95% Confidence Interval
Psize 500	13.980	6.941	12.36	[13.153, 14.806]
Psize 1000	12.902	5.406	12.98	[12.195, 13.610]
APS	12.053	4.750	11.56	[11.226, 12.888]

Table 4.2 presents the mean best fitness and standard deviation over 200 runs for the three scenarios of fixed population size 500, 1,000, and APS. Clearly, APS

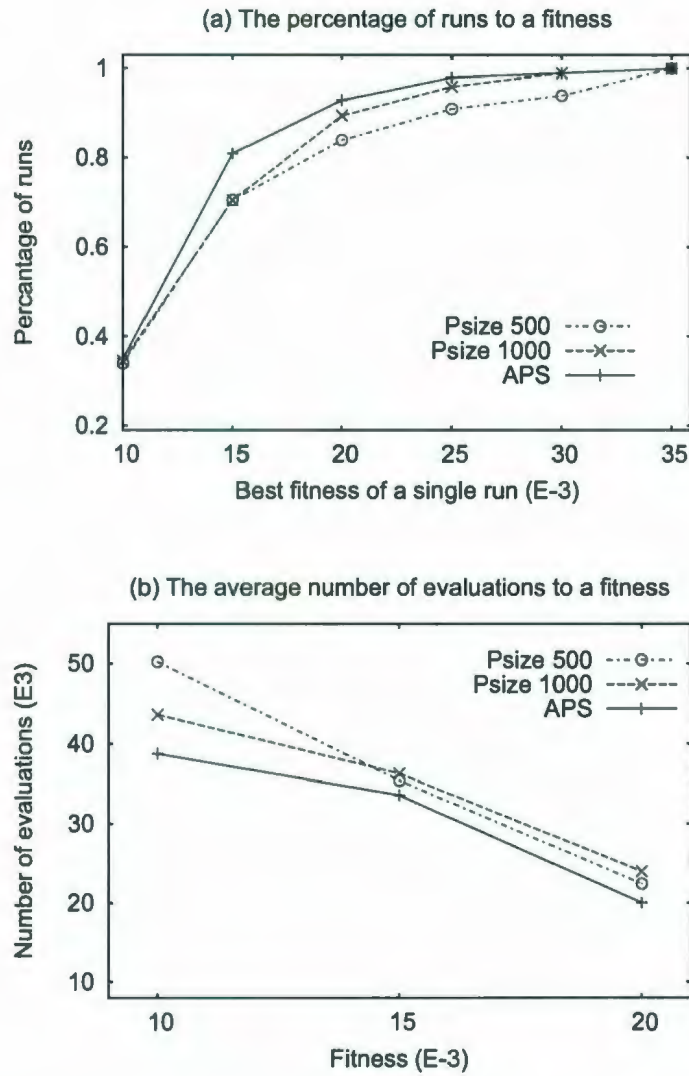


Figure 4.6: Solution quality and computation costs comparisons.

achieves an even better fitness than maintaining 1,000 individuals but its population size is mostly between 500 and 1,000.

The cumulative success rates of these three groups are depicted in Figure 4.6(a). We focus on the best fitness of a run once it terminates. The figure plots the percentage of the total 200 runs of each scenario that yield a better fitness than a given

threshold between 0.01 and 0.035. Apparently, APS has the highest percentage for all of the cases, which indicates its superiority over fixed population size strategies. Figure 4.6(b) reveals a dual measure for the three scenarios. In this chart, we compare the average number of evaluations needed for a simulation run to achieve fitness levels of 0.01, 0.15, and 0.02. For a given fitness level, APS always incurs less computation overhead, and this difference becomes greater as the fitness requirement gets higher.

4.4 Discussion

The role of population size for rate of evolution in a GP system is investigated in this chapter. We transferred an idea from population genetics that population size has varying effects on the rate of evolution under differing selection regimes. The measurement k_a/k_s ratio was revised and adopted as an indicator for population size adjustment. We proposed and tested that dynamically adjusting population size can effectively stabilize the rate of genetic substitutions even during late stages of an evolutionary process. It has been further verified that this strategy can also successfully accelerate the rate of fitness progression.

The adjustment of population size is indeed an operation to vary the selection pressure on accepting neutral or nearly neutral genetic changes. When the evolution of a system is stagnant, encouraging neutrality exploration can be an effective way to provide variation potential since the search space is enlarged. We also believe that neutrality contributes a great deal in evolvability. This perspective will be studied in the next chapter.

Chapter 5

Neutrality and Variability: Two Sides of Evolvability

It would be overly optimistic to expect a formula to describe evolvability mathematically, due to the complexity of organisms, the dynamics of populations, and the influence of the environment. The most striking feature of evolvability is its capability to generate *adaptive* phenotypic variation from *random* genetic changes. Neutrality and variability are the two sides of evolvability important in controlling random genetic changes.

Genetic changes do not necessarily result in any observable phenotypic variation. This “neutrality” has two functions, i) it improves a system’s robustness against deleterious genetic changes, and ii) it provides variation potential through exploiting neutral networks [15, 40]. In contrast, “variability” generates observable phenotypic variations for adaptation to the environment. These two sides of evolvability may appear contradictory at first sight. However, they closely cooperate to facilitate

evolution. Moreover, it is the environment that dictates which side dominates at which stage of an evolutionary process.

In this chapter, we investigate evolvability as the exploration of its two sides, neutrality and variability, under various environmental scenarios. The rate of evolution measurement k_a/k_s is adopted as a very important index to observe the temporal aspect of this progress. Since the GP system is chosen in our simulated studies in previous chapters, we stick to GP as our test case but adopt the linear-coded GP representation here, focusing on a polynomial symbolic regression problem.

5.1 Evolvability and environment

It has been well accepted that evolution can be understood in general with three fundamental elements: variation, selection and inheritance. It is impossible to study a system's evolutionary capability without consideration of its environment. The detection and investigation of evolvability are non-trivial and intriguing problems. Phenotypic fitness is directly observable and serves as a selection criterion. However, as a *potential* to generate better fitness and a *capability* for adaptive evolution, evolvability is more difficult to observe and to select for. Therefore, some empirical methods have been proposed in the literature to investigate evolvability "indirectly" in various environments.

Orr [133] analyzes the acceptance of mutations in a system moving toward a stationary optimum, and suggests that the effects of accepted mutations are decreasing. That is, towards a fixed target, rapid phenotypic variation can be observed at the beginning, but the rate of observable adaptive evolution will slow down later on.

Further, Collins et al. [30] study adaptive walks in dynamic environments, and report that the rate of environmental change has a systematic effect on adaptive walks. Gradual changes allow more small-effect genetic substitutions than sudden changes, and favor more robust individuals that do not behave poorly in any intermediate environment. Thus, a large drop in fitness almost never happens in a gradually changing environment. Earl and Deem [39] suggest that evolvability can be selected for by varying the environment. By observing genetic changes in protein evolution, they find that rapid or dramatic environmental changes generate strong selection pressure for evolvability. Thus, high evolvability can be detected and favored by such selection pressure. Meyer et al. [118] state that fluctuating environments can drive populations towards the edge of a neutral network. Kashtan et al. [85, 136] report that varying environments, especially in a mode that prefers modular changes, can facilitate rapid adaptive phenotypic variations.

In the GP literature, evolvability has also emerged as a very important research topic. Ebner et al. [40] incorporate redundant mapping from genotype to phenotype in evolutionary computation models as a form of neutrality and show how neutral networks can influence evolvability. Further, Banzhaf and Leier [15] examine the behavior of an evolutionary search process in neutral networks using Linear GP for a stationary Boolean search problem. Belle and Ackley [23] design a dynamic environment exploiting modularity among varying goals, and argue that enhancing the search modularity in a changing environment can increase a GP system's evolvability. Yu [188] reports that GP populations exhibit various program distributions under different environmental variation rates.

Inspired by those interesting discoveries from Biology and motivated by the im-

portance of studying evolvability in GP, we are interested in observing those biological discoveries on GP to see if similar findings can hold in our computation system regarding those general principles of evolution.

We focus on the influence of the environment on the two sides of evolvability, neutrality and variability [76]. Variation is the driving force of evolution. However, most random genetic variations are well known to be deleterious. Evolutionary systems exploit neutrality and variability, as two opposite strategies, to control random genetic changes in different situations. The core of evolvability is to generate *adaptive* variations at the phenotypic level from random genetic changes. Therefore, we believe that the cooperation of persistence and sensitivity to random genetic changes contributes substantially to evolvability, and the dominance of either side is driven by the environment. It is hypothesized that evolvability of a computational evolutionary system can also have different exhibitions exploring its two sides in various environments.

5.2 Methods

Two methods, with one emphasizing the temporal aspect and the other emphasizing the spatial aspects of evolvability, are adopted. The nonsynonymous to synonymous substitution ratio k_a/k_s captures how neutrality and variability interplay with each other with time. The neutral networks depict which side of these two dominates at particular points of time, that is, for all the individuals in a GP population at a certain generation, whether they are very robust or very sensitive to genetic changes.

5.2.1 The k_a/k_s ratio for Linear GP

As introduced in previous chapters, in the k_a/k_s ratio measurement, k_a describes the rate of nonsynonymous substitutions and k_s measures the rate of synonymous substitutions. This measurement practically captures how the relative importance of either neutrality or variability changes with time.

We slightly adapt the k_a/k_s ratio measurement for Linear GP. Note that there can be variants in defining nonsynonymous and synonymous genetic changes. For a strict analogy to biological systems, which often have no easily measurable fitness, the effects of a genetic change would refer to the influence on its phenotype. However, in a GP system, fitness is explicitly defined and is in most cases the only criterion for selection. Therefore, we make the simplifying assumption that a GP genetic change is nonsynonymous (synonymous resp.) if it changes (maintains resp.) the fitness of an individual. Other parts of the definition of the k_a/k_s ratio stay the same as previously described.

5.2.2 Neutral networks

In genotype space, a neutral network is usually defined as a set of genotypes that map to the same phenotype [15, 40, 118, 176]. Each genotype corresponds to one vertex in the neutral networks. A genotype G_1 is linked to another genotype G_2 if G_2 can be obtained from G_1 via a one-step mutation. Note that these links are usually bidirectional due to the reversibility of mutation. Further, a link can exist both within and across neutral networks. We say that a genotype is a “neighbor” of G_1 if it is linked with G_1 . In addition, it is a “neutral neighbor” if it belongs



Figure 5.1: An example of neutral networks.

to the same neutral network as G_1 . Otherwise, it is a “non-neutral neighbor”. For a given genotype, we follow Wagner [176] in defining its variability as the fraction of non-neutral neighbors among all of its neighbors. This quantifies the likelihood that a mutation from a given genotype leads to a phenotypic change. Again, as a simplification in this contribution, we assume that two genotypes are in the same neutral network if they have the same fitness, instead of looking at their phenotypes.

Figure 5.1 depicts a simple example of three genotype neutral networks. Black lines show the connection within a neutral network, and grey lines mark the connection among different neutral networks. Genotypes with high variability are positioned near the edge of a neutral network, and genotypes more robust against genetic changes are placed close to the center of this network. Therefore, the distribution of individuals in neutral networks can reflect the relative importance of either neutrality or variability of a population at a given point in time.

For simple problems, all reachable genotypes can be exhaustively enumerated. However, the genotype space grows exponentially with the complexity of a problem. Thus, we need to sample the genotype space to obtain an approximation for complex

problems. That is, for a given genotype, we sample a sufficiently large number of its neighbors to estimate its variability. This is the approach we will adopt here.

5.3 Simulated studies with Linear GP

We use Linear Genetic Programming in our experiments. We choose Linear GP over the more commonly studied Tree GP because Linear GP seems to have a better resemblance to biological systems. Moreover, we would like to study a different representation since we have tested the k_a/k_s ratio on a tree-based GP system previously. We design a set of varying environmental scenarios, and measure the k_a/k_s ratio and the variability of genotypes in neutral networks in order to investigate evolvability in different environmental situations.

5.3.1 Test case: polynomial regression

Our benchmark is the polynomial symbolic regression problem ($\sum_{i=1}^n x^i$, for some n). Note that there can be similar patterns within this polynomial. For example, when $n = 4$, $x^4 + x^3 + x^2 + x = x(x + 1)(x^2 + 1) = x^2(x^2 + 1) + x(x^2 + 1)$. Also, if we increase n , we can design moving targets based on this expression. Here, only mutation is used for genetic changes. Each mutation can take two forms. A micro-mutation limits the change to one element of a specific instruction, i.e., the return register, the operator, or one of the two operand registers. A macro-mutation inserts a randomly generated instruction into the program or deletes one instruction, either at a random location. In particular, the mutation rate of a program is 1, with half of the likelihood happening at the micro level and half at the macro level. When a

Table 5.1: Linear GP configuration for polynomial regression.

Population size	1000
Initial program length	5 – 15
Maximum program length	200
Number of input register	1
Number of calculation register	3+1
Constants	1,2,...,9
Operator set	+, -, ×, protective ÷
Mutation rate	1
Sample set	100 cases in [-1:1]
Fitness function	Mean error
Truncation selection	Tournament with size 6
Maximum generations	500
Neutral network space	1000 sampled neighbors

program adopts a macro-mutation, the instruction insertion and deletion occur with equal probability. We employ a truncation selection scheme where both parent and offspring populations will compete to form the next generation. The configuration is specified in Table 5.1.

5.3.2 Varying environmental scenarios

Here, in the context of a Linear GP system with symbolic regression, we define its environment as the target polynomial expression. In this sense, typical environmental

scenarios include i) random evolution, where no specific evolution target is defined, ii) fixed target evolution, and iii) moving target evolution. In the following, we study the effects of these scenarios on Linear GP.

5.3.2.1 Random evolution (RE)

We implement random evolution by applying random selection when forming a new generation. We plot various measurements of the process in Figure 5.2. In Figure 5.2(a), we plot the average program length over time and observe that there is no general trend of length. This is distinct from normal Linear GP, where average program length increases. All other metrics in this figure, however, indicate a fair level of stability (Figure 5.2(b)-(f)). In Figure 5.2(b), the system presents a consistent 20-80 split among the 1000 total mutations between the nonsynonymous (N_a) and synonymous (N_s) changes. The accepted nonsynonymous (M_a) and synonymous (M_s) substitutions remain at half of the level (Figure 5.2(c)) because a new generation always starts with the combination of all parents and offspring and half of them survive at random. This means that the k_a and k_s rates are both approximately 0.5 (Figure 5.2(d) and (e)), with k_a having slightly higher variance. This further implies that the k_a/k_s ratio stays at around 1, i.e., the neutrality and variability apply equal influence in a random evolution system.

In addition to observing the system as it progresses, we are also interested in the variability of all the individuals at time snapshots. In particular, we plot our measures at the beginning and end of the evolution in Figure 5.5(a). In the figure, each snapshot corresponds to one plotting. Numbers in the parentheses represent a typical generation. We sort the individuals according to their degree of variability

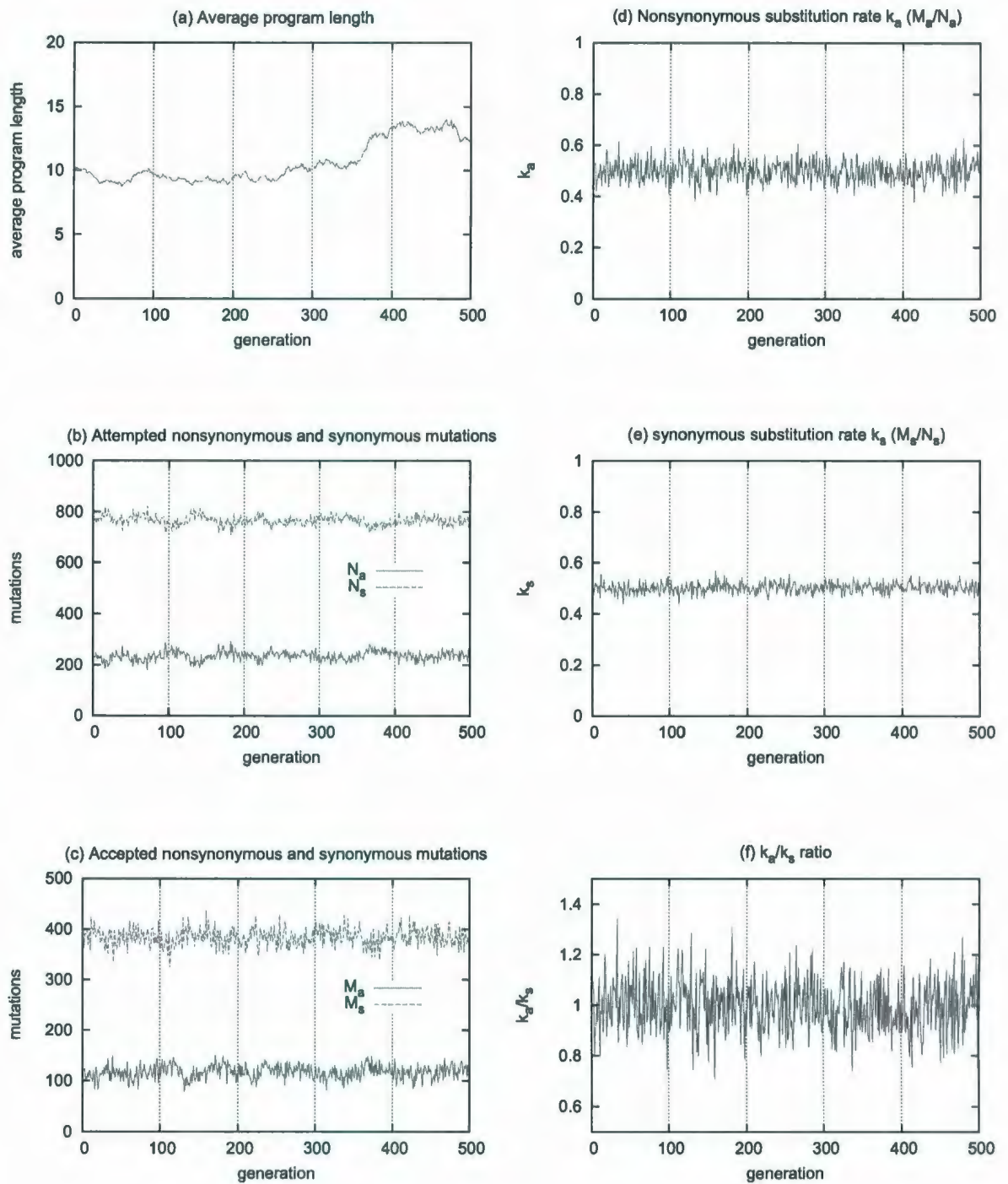


Figure 5.2: A typical single run of random evolution.

for better readability. Apparently, the random evolution process does not alter the variability composition of the population.

5.3.2.2 Fixed target (FT) evolution

In this experiment, we start out with a simple fixed target of $x^2 + x$. The evolution quickly leads the first individual to optimum at generation 5 and the entire system converges to this optimum at generation 10. After this, the average program length keeps increasing (Figure 5.3(a)), which builds more and more redundancy into individual programs. Figure 5.3(b) records the number of nonsynonymous mutations N_a and that of the synonymous mutations N_s at each generation. These two metrics start with a 20-80 split as with the previous scenario because of the randomness of the initial population composition. As the system progresses towards the optimum and converges (up to generation 10), N_a increases as a large number of the mutations are nonsynonymous. After this point, N_a decreases and approaches 0 due to the increasing robustness in the population. N_s follows the complementary trend in this process. As in Figure 5.3(c), the system starts to completely reject nonsynonymous changes (M_a) after the convergence to the optimum because any such change is deleterious and is not favored by selection. During the process, M_s remains at about half of the level of N_s because of the half-half composition of a new generation before selection. The nonsynonymous substitution rate k_a (Figure 5.3(d)) has a positive value until convergence, indicating no phenotypic evolution occurs after this point. The synonymous substitution rate k_s (Figure 5.3(e)) suggests a very active background evolution before system convergence, which stabilizes at approximately 0.5 afterwards. As a result, the k_a/k_s is always less than 1, and has a positive value

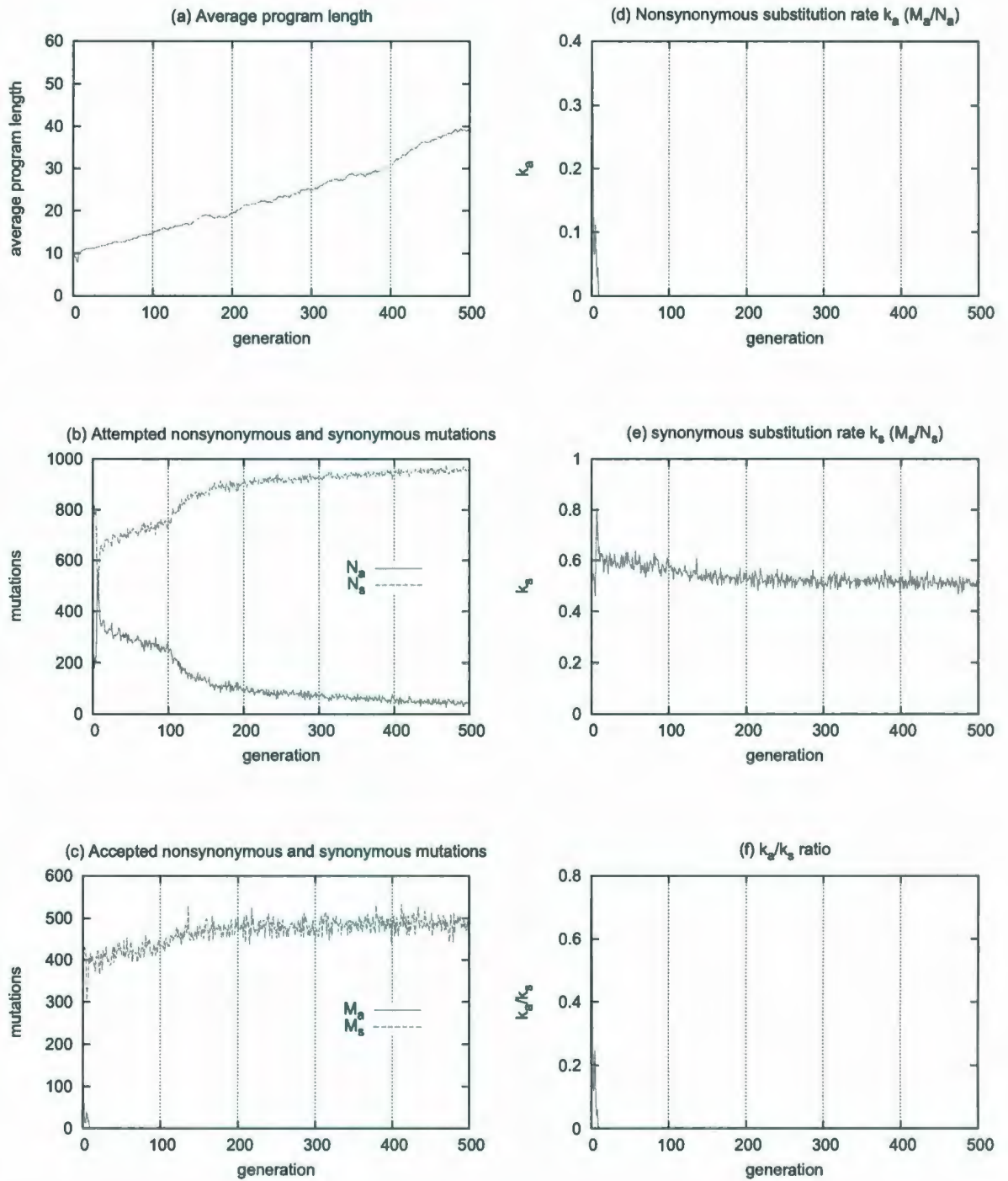


Figure 5.3: A typical single run of fixed target evolution on $x^2 + x$.

until the system converges. This is the result of the majority of random mutations being deleterious, which is a recognized phenomenon in both biological and artificial evolutionary systems.

We next zoom in to the results of quartic polynomial regression ($x^4 + x^3 + x^2 + x$). Compared to the simpler target of $x^2 + x$, the evolution here takes a longer time to complete, but the general trend of these two runs is the same. The quartic case provides more abundant information to study the process of locating various local and global optima. Here, we plot only the first 200 generations during evolution (Figure 5.4). The best fitness and average fitness are plotted in Figure 5.4(a), where the fittest individual hits the global optimum at generation 117 and the fitness converges at generation 123. Notice that there is approximately 5 generations of lag between hitting a local or global optimum and assembling the population to that point. Figure 5.4(b) plots the number of individuals that have the same fitness as the fittest individual over time. Observe that there are 4 periods of frequent replacement of the fittest individual, i.e., generations 10-15, 70-75, 85-90, and 110-120. As with the previous scenario, we also plot the mutations (Figure 5.4(c)), accepted substitutions (Figure 5.4(d)), and their relative rates (Figure 5.4(e)). In all these measurements, we observe whenever there is frequent replacement of the fittest individual, the system is actively yielding and accepting phenotypic variations. Note that the rate of k_g remains at approximately a constant level regardless of the system dynamics. However, k_a faithfully captures the rate at which the system makes observable improvements. Thus, the ratio of k_a to k_g also provides a reliable measurement of evolution rate. Consequently, all of the metrics shown in Figure 5.4 verify that alternation of the dominance of neutrality and variability is a driving force for evolution throughout

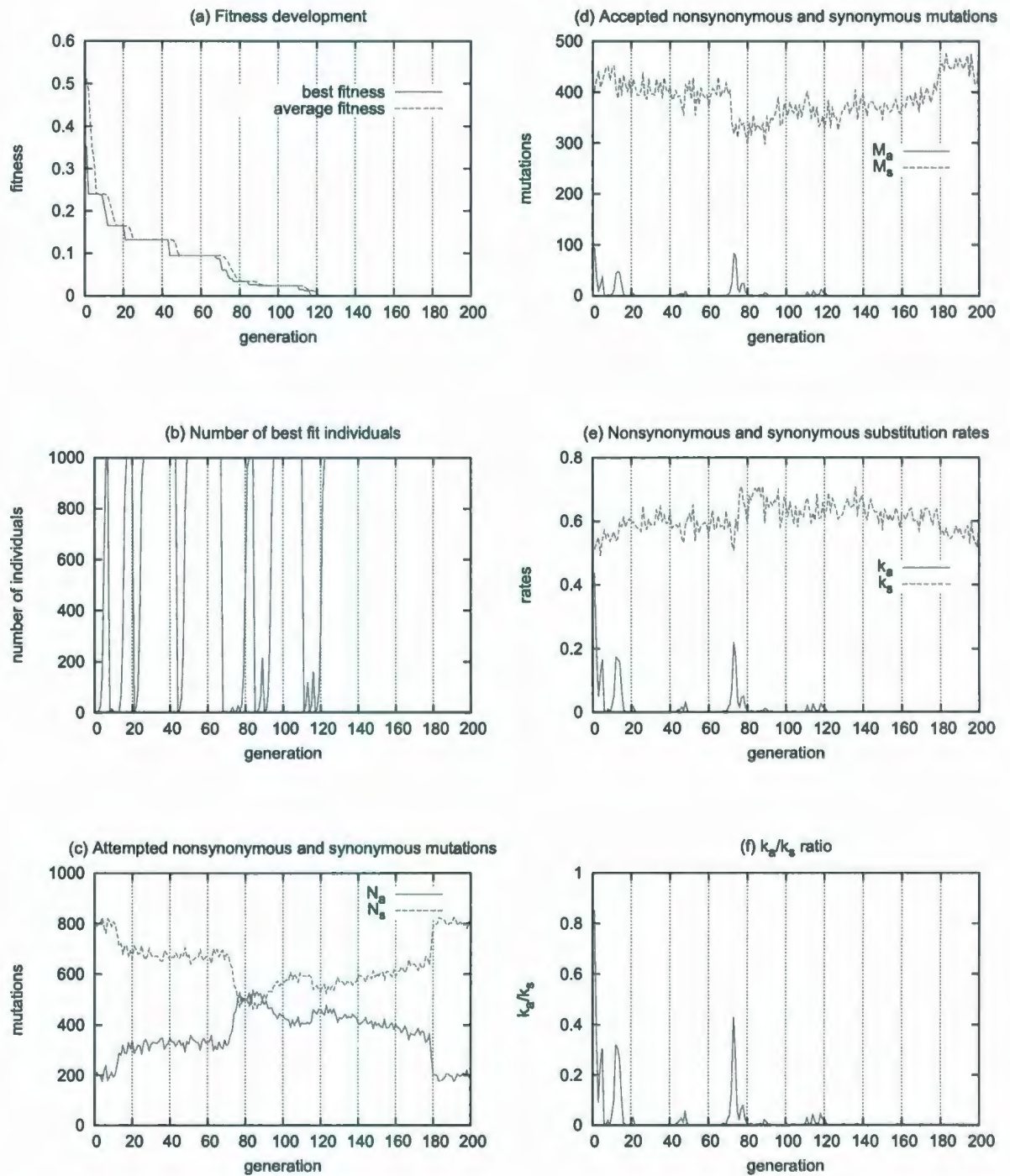


Figure 5.4: A typical single run of fixed target evolution on $x^4 + x^3 + x^2 + x$ for the first 200 generations.

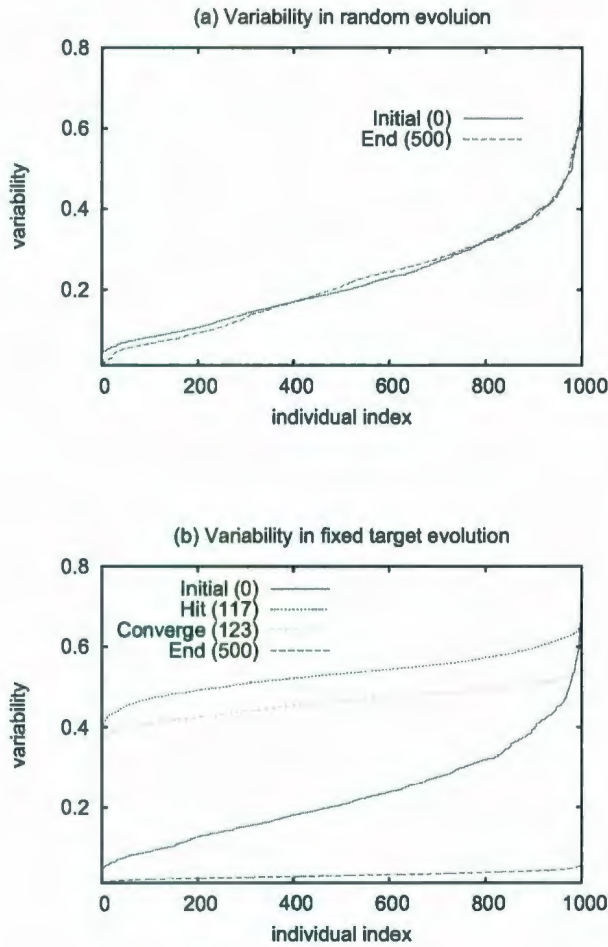


Figure 5.5: Variability measurements in random evolution and fixed target evolution.

time. In addition, the observations we have made here coincide with biological evolution in that i) most random mutations are deleterious so that the k_a/k_s ratio is mostly less than 1, and ii) this ratio generally decreases as fitness improvements become finer-grained [133].

In terms of system variability (Figure 5.5(b)), we are interested in four points of time during the evolutionary process. That is, at the very beginning, when the fittest individual hits the optimum, when the system converges, and at the end. We

observe that the initial population possesses the same high diversity in variability as in random evolution (Figure 5.5(a)). As the system evolves, the population has a high overall level of variability but less in diverse. When the system converges to the optimal fitness and the population starts to possess approximately the same genotypic structure, both the variability and diversity decrease, but the system is still fairly sensitive to mutations. As the evolution progresses to the end of the run and more redundancy accumulates, the entire population has very low variability eventually.

In both the temporal and spatial sense, when a system has a specific target posed by the environment, the coordination between neutrality and variability behaves rather differently from void environmental influences.

5.3.2.3 Moving target (MT) evolution

We design a moving target by increasing the degree n of the polynomial $\sum_{i=1}^n x^i$ periodically. Thanks to the similarity among these targets, there is a good amount of inherent modularity in these environmental changes [85]. In the following experiments, we study how the system responds to such modular changes. At the outset, the system evolves towards the polynomial $x^2 + x$, but we change the target to a higher degree every c generations, called the *switching period*. Specifically, the target is a function of time (or generation) t ,

$$T(t) = \sum_{i=1}^{\lfloor t/c \rfloor + 2} x^i. \quad (5.1)$$

Here, we change the target polynomial every 100 generations ($c = 100$), and then allow our Linear GP system to evolve for 500 generations. The target polynomial will

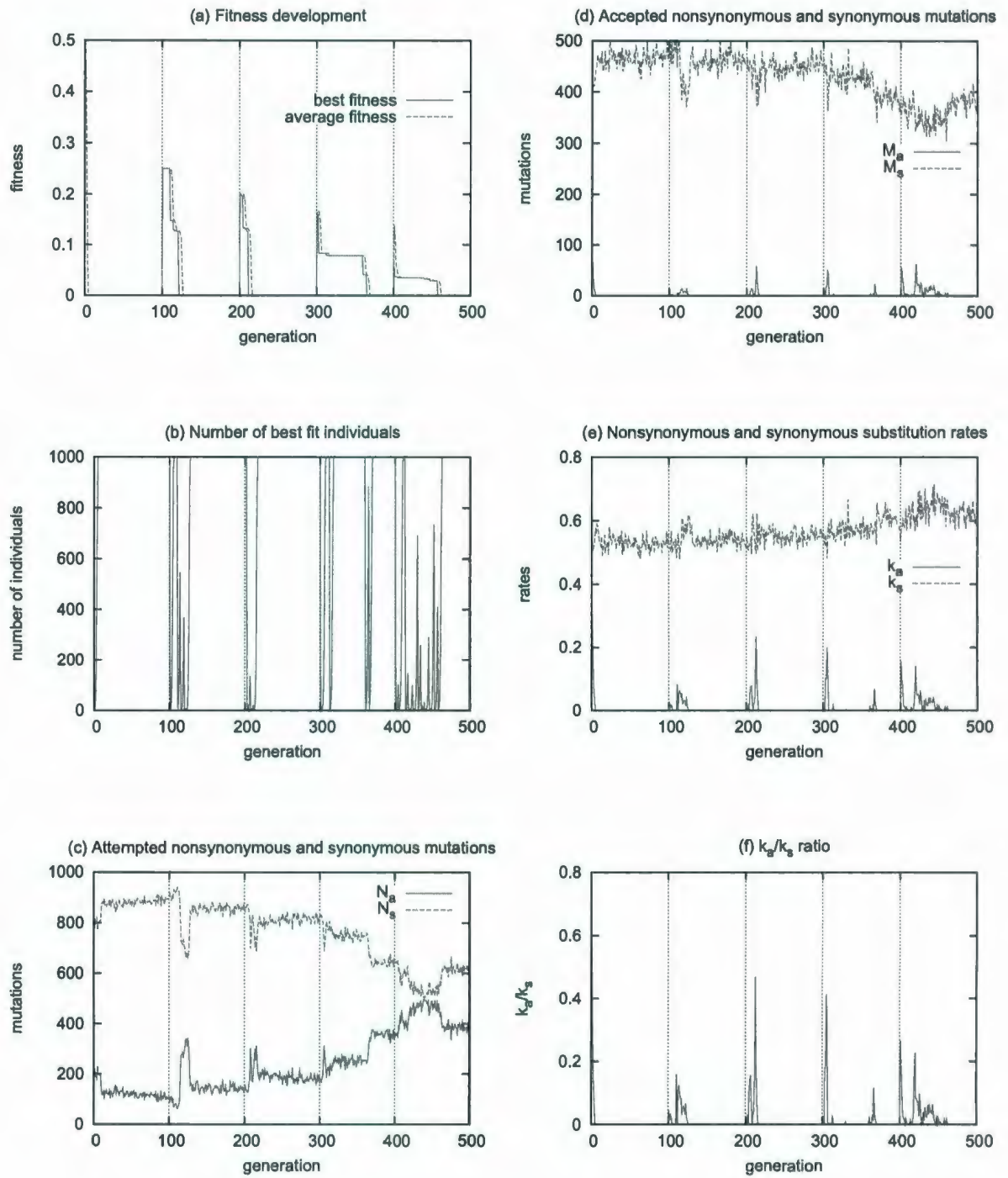


Figure 5.6: A typical single run of moving target evolution.

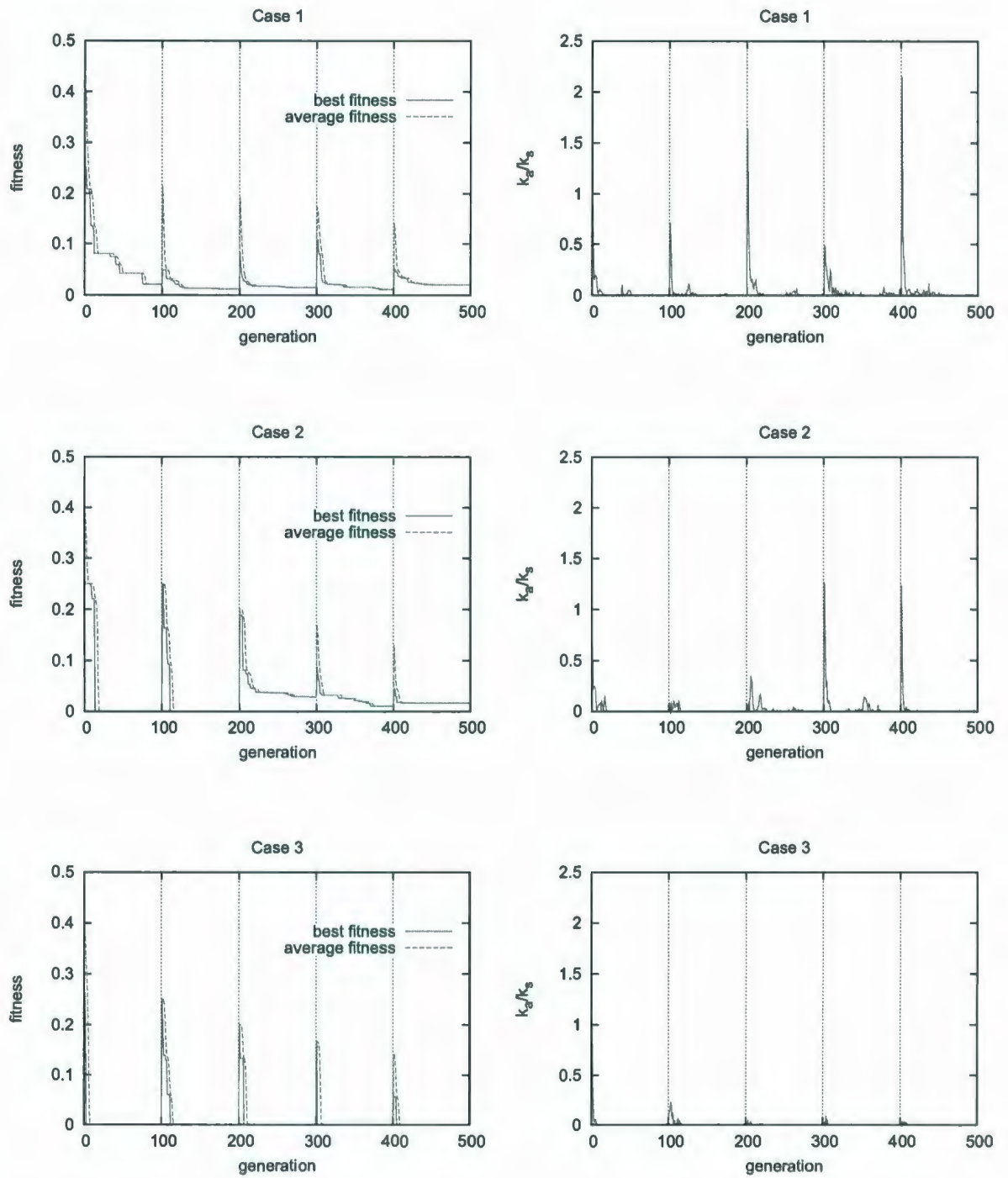


Figure 5.7: Three typical cases of moving target evolution.

increase its degree from 2 to 6. As the polynomial degree increases, the target takes a more complex form. However, since the target changes in a modular way, there can be many reusable patterns from previous target polynomials, and the search process is expected to learn from history.

Figure 5.6 depicts the metrics we looked into in previous scenarios. Each time the target is switched, we see that the fitness worsens (Figure 5.6(a)) and the replacement of the fittest individual happens frequently (Figure 5.6(b)). This is similar to the fixed target scenarios, where individuals are becoming sensitive to mutations whenever the system is frequently replacing its currently fittest individual (Figure 5.6(c)(d)). It is interesting to see from the chart for the nonsynonymous and synonymous substitution rates (Figure 5.6(e)) that, despite the periodic target switching, the synonymous rate k_s still stays fairly stable. This indicates that neutral genetic changes take place and are accepted at a stable rate during the entire evolutionary process, but phenotypic variations can only be observed when the system adapts to its new environment. Again, this results from the close cooperation of neutrality and variability, as two sides of evolvability harnessing random genetic changes to generate adaptive phenotypic variations.

Moreover, carefully designed modular target switching is expected to accelerate evolution. We present three typical runs in Figure 5.7 to investigate this. For each case, we plot fitness development and the k_a/k_s ratio. In case 1, the system cannot reach the target in any period before the target moves. As discussed previously, in this case the system is changing very actively. In case 2, the system only finds and converges to the target for the first two periods. In case 3, the system successfully reaches the target by the end of each period. In all of these cases, we observe that,

when the target is moved before the system finds and converges to it, the fitness changes are smaller at the target switching point and the k_a/k_s ratio is higher at these points. In contrast, the system is slower to respond to a target change if it has found and converged to a target previously. In terms of neutral networks, as the system finds and converges to a target, the individuals of the system “settle to the center” of the neutral network, and the system becomes more robust. Thus, phenotypic variations start slowly once it is exposed to new environmental challenges. In this case, the individuals need to first move to the edge of the neutral network, i.e., to “pull” the system out of stagnation, before adapting to this new environment. Another observation is that the polynomial target changing in a modular way can improve search efficiency. That is, an evolutionary system can find an ultimate target by following a series of intermediate goals faster than by trying to find it directly. This also suggests some interesting future research on problem modularity and evolvability.

5.4 Discussion

The most important feature of evolvability is its capability to generate adaptive phenotypic variations from random genetic changes. Neutrality and variability are the two sides of evolvability controlling random genetic variations. The environment plays an important role in evolvability to determine which of these two sides is dominant. In this chapter, we employed a Linear GP system as a case study to examine the behavior of evolvability in various environmental situations, by using two tools that can capture evolvability in the temporal and spatial senses. We observed that an evolutionary system actively generates phenotypic variation only when it is adapting

to a new environmental challenge. However, this adaptivity is not coming out of void but is the result of constant genetic variations in the background, with the majority being neutral. To cope with environmental fluctuations, a system can improve its phenotypic variation rate without changing its genetic variation rate.

It is quite rewarding to observe behaviors similar to those seen in natural systems. This work also helps us attain a better understanding of the general principles of evolution, suitable to both natural and simulated computation systems. We also would like to highlight the following observations from this study. First, neutrality is very important. It provides not only the protective robustness in an evolutionary system, but also the future variation potential. Those neutral or nearly neutral variations contribute a great deal in the search process. They provide a hidden staging ground for future phenotypic changes. Second, a changing environment is crucial to studying evolvability. An evolutionary system does not have to make changes if it is highly adaptive to its current surrounding environment. The phenotypic evolution is only observable when this system is changing against certain environmental selection pressure. Therefore, high evolvability lies in the capability of adapting to a varying environment. Although we only adopted a simple environment changing scheme in this work, we speculate that the intensity, the rate, and the pattern of environmental changes can have considerable impact on evolvability.

Chapter 6

Real-World Application

Applying evolutionary algorithms to real-world problems is important to test an algorithm's design and thus to impel the improvements of the algorithm to achieve better performance. In this chapter, a wireless network planning problem is adopted as our application.

IEEE 802.16, also known as WiMAX, is a new wireless access technology for currently increasing demand of wireless high-speed broadband service. Efficient and effective deployment of such a network to service an area of users with certain traffic demands is an important network planning problem. This network planning can be formulated in a similar way as the unsplittable capacitated facility location problem. Different from traditional p -median or splittable capacitated facility location models, the unsplittable capacitated facility location problem is even harder. In addition, the limited communication range adds another layer of complexity.

Here, we resort to an evolutionary approach in order to yield good approximation solutions. In our method, individual representation and genetic variation opera-

tions are specifically designed to incorporate the features of this application problem. Moreover, the rate of evolution measurement proposed in Chapter 3 and the adaptive population size approach to enhance neutral search proposed in Chapter 4 are further tested on this particular application problem.

6.1 Wireless network planning problem

WiMAX (Worldwide Inter-operability for Microwave Access) is a telecommunication technology based on the IEEE 802.16 standard in order to provide broadband wireless networks at the metropolitan scale. It intends to replace the more expensive wireline-based access technologies such as TV cable and ADSL [43, 102]. As the standard evolves, WiMAX supports a variety of data transmission methods in the 10-66 GHz and 2-11 GHz spectrums. It originated from the first 802.16 standard in 2002, also called WirelessMAN, where a cellular-like *point-to-multipoint* (PMP) operation is adopted. In the PMP mode, all communications are limited to be between a basestation (BS) and a subscriber station (SS). In an amendment in 2003, 802.16a, a new operation mode of *mesh* was added to allow direct communication between SSs. In an later amendment in 2005, IEEE 802.16e added mobility extension to the previously fixed WiMAX. Currently, a new working group, 802.16j, is focusing on multi-hop extensions so that the network can operate in a *mobile multi-hop relay* (MMR) mode. With the relay stations (RSs) to help, the coverage of the BSs can be increased significantly, which alleviates the line-of-sight (LOS) problem further [55].

6.1.1 Problem description

Here we focus on the PMP mode of WiMAX, where there can be two types of entities to form the wireless component of the network, the BSs and SSs. The BSs form the infrastructure for the SSs. An SS is allowed to communicate to a BS directly if the channel quality is sufficient for the given data rate. A network planning problem in this case is an optimization problem to cover the SSs in a geographical area using a small number of BSs. The BSs can only be placed on a subset of pre-selected candidate sites. Typically, the locations of the SSs and their bandwidth requirements are given. In addition, the channel gains between the locations of the SSs and all BS candidate sites can also be obtained. Thus, for a given candidate site, the set of SSs that can be serviced by this site is known as well. Note that, in practice, since every BS has a capacity upper limit, it may not necessarily service all these SSs within range. We assume that there is no power control mechanism at either end of the channel. An example is provided in Figures 6.1 and 6.2. Figure 6.1 is an instance of 11 users within the range of 6 candidate sites. Assuming that each candidate site can service up to 3 users. Thus, the diagram in Figure 6.2 is a solution of using 4 BSs to construct the infrastructure. In this particular example, this happens to be the only solution, but generally the number of solutions can be exponentially large.

6.1.2 Problem formulation

The *network planning problem* can be modeled as a minimization problem on a weighted graph $G = (V, E)$. Specifically, there are two types of vertices in the graph, i.e., $V = B \cup S$, where B represents the candidate basestation sites and S represents



Figure 6.1: Example of a wireless network layout.

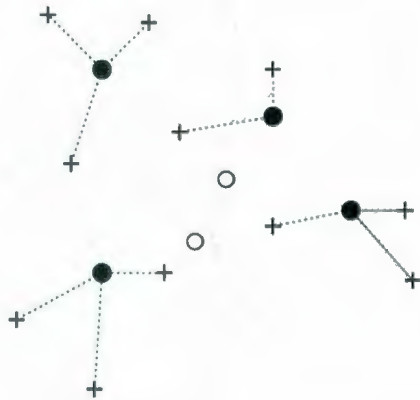


Figure 6.2: Example of a network planning for the network layout in Figure 6.1.

the subscriber stations. For each $s \in S$ and $b \in B$, there is an edge between them if the channel gain $g(s, b)$ between s and b is greater than or equal to a given threshold δ for data reception. Therefore, graph G in this case is a bi-partite graph, where there are no edges within B or S themselves. Every $s \in S$ is associated with a capacity requirement of bandwidth c_s . The candidate basestation sites each have a capacity

limit of C , which caps the total amount of bandwidth of its connected SSs.

A *feasible plan* is a mapping $M : S \mapsto B$ that satisfies the following constraints.

1. For each $s \in S$,

$$g(s, M(s)) \geq \delta. \quad (6.1)$$

2. We define the *load* of a BS $b \in B$ as

$$l(b) = \sum_{M(s)=b, s \in S} c_s$$

and enforce a capacity limit on it, i.e.,

$$l(b) \leq C. \quad (6.2)$$

The total infrastructure cost of the network lies in the number of BSs in use. Therefore, our goal is to minimize $|M(S)|$ over all feasible plans.

Network planning as an optimization problem, in different flavors, has attracted research interest recently. When a BS has a capacity limit, the problem is called *capacitated*; otherwise, it is *uncapacitated*. Amaldi et al. studied the problem in uncapacitated UMTS cellular networks by formulating the problem as an Integer Program (IP) [7]. It is assumed that the nodes are able to change their transmission power adaptively. Thus, the objective is to minimize the total cost of operating a number of BSs and of transporting data from the SSs at an appropriate power level for sufficient reception gain. To solve the NP-hard IP, they resorted to randomized greedy search and tabu search. In Yu et al. [190], a two-tier assignment variant is considered to model 802.16j MMR, and the RSs and BSs are uncapacitated. In their solution, a fixed number of BSs is considered so that the top-level assignment can be

treated by a p -median clustering. Generally, when an SS is allowed to be serviced only by one BS (or RS), we say the problem is *unsplittable*, as in the work discussed above. Alternatively, with more sophisticated scheduling and channel assignment, an SS may be serviced by multiple BSs (or RSs) equivalently. This is called *splittable*. In Lin et al. [104], a flow-based heuristic is devised to relax the capacitated IP formulation essentially to a splittable variant. This is a generalization of the problem of *capacitated facility location* [147], where an SS can be potentially serviced by all BSs with different transportation costs. The variant that we consider in this work is the more difficult unsplittable capacitated network planning problem, where a BS can only service the SSs within range. When user demands are not allowed to be split, flow-based solutions are not useful any more.

From a broader context of combinatorial optimization, it is important to understand that the network planning problem is considerably more difficult than the better studied bin-packing and p -median problems. Here, because each BS has a different set of users in range, they are not equivalent in terms of capability of servicing the users. This is distinctive from bin-packing where all bins are equal in pursuit of using a minimum number of them. Compared to p -median, here we can use a varying number of BSs to satisfy the users rather than a fixed number p . In the EC community, there is an increasing need for customizing evolutionary methods to closely incorporate the features of the combinatorial optimization problems [4, 32, 45, 117, 144].

6.2 Evolutionary approach to network planning

For constrained combinatorial optimization problems, genetic variation operations in evolutionary algorithms are usually destructive to invalidate an individual as a candidate solution. Simply applying general and conventional genetic operations without specific heuristics could not be able to exploit the automatic search power of evolutionary algorithms. There is an increasing need for customizing evolutionary methods to closely incorporate the feature of a problem. Furthermore, the rate of evolution measure k_a/k_s ratio and the adaptive population size approach proposed in previous chapters are applied here. This helps to verify if an evolutionary algorithm can benefit from the central idea of this thesis, i.e., enhancing neutral search in a system's evolvability, when working on a real world application.

The framework of our evolutionary approach is described with a view on four specific aspects [78]. We start out with a description of how to represent a solution to the network planning problem using a two-tier genetic structure in order to encode the BS selection and SS assignment separately in Section 6.2.1. Next, we outline the iterative genetic operations applied to the population to approach the optimum in Section 6.2.2. Then, Section 6.2.3 explains the incorporation of our adaptive population size scheme in this application. Note that the fitness of an individual is defined as the number of BSs in service. Thus, there can be many tied solutions with the same fitness but not necessarily the same set of activated BSs and associated SSs. Although this neutral diversity is not observable at the fitness level, it plays an important role in expanding the genotypic search space. The adaptive population size scheme allows a system to dynamically enhance neutral search during different stages

of the evolution by population size adjustment. Last, in Section 6.2.4, we explore evolutionary operations, including crossover, mutation and a repair heuristic, in the spirit of the network planning problem.

6.2.1 Individual representation

Given a set of subscriber stations S and a set of basestations B with their location information, we encode a mapping M from S to B as a two-tier chromosome. At the higher level, i.e., the *BS activation level*, we use an array of length $m = |B|$ to represent the BSs. In addition, each locus i of this chromosome stands for a BS b_i , referring to its service list containing all the SSs assigned to it. If there is no SS connected to a BS (i.e., this BS is not needed), its service list is \emptyset . This is referred to as the *SS assignment level*. Such a two-tier representation is depicted in Figure 6.3.

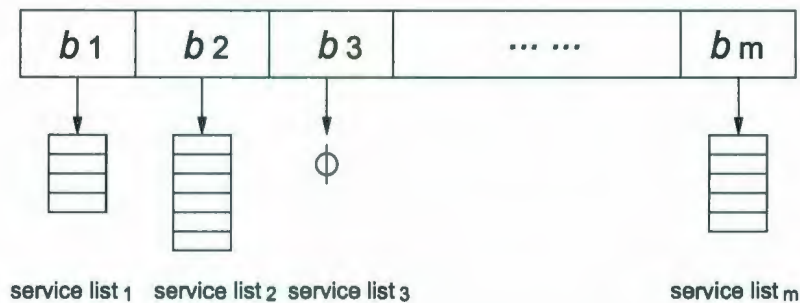


Figure 6.3: Two-tier chromosome representation.

For a feasible solution, the total length of the service lists should add up to $|S|$, and for each b_i the total capacity demand in the list must not exceed the BS capacity. Our goal is to minimize the total number of loci referring to non-empty service lists. The division of information into two tiers separates the semantics embedded in an

individual. That is, the activation of a BS and the assignment of an SS to an activated BS are encoded in two separate domains. This allows us to control the genetic variations at these two levels independently, which turns out to be fairly powerful as indicated by our experiments. This two-tier genotype is distinctive from the most common representations of GA solutions to combinatorial optimization problems. In such works, the genotype usually takes a fixed form to resemble a biological gene sequence. In particular, a genotype would consist of $|S|$ loci, each of which refers to the index of the BS servicing this SS. Alternatively, in the fixed-structure genotype camp, a genotype would represent a solution by an indicator matrix $\{0, 1\}_{|M| \times |S|}$, where each column i ($i = 1, 2, \dots, |S|$) contains exactly one 1 and $|B| - 1$ 0's. One noticeable exception to this is the "multi-level encoding" in Meunier et al. [117]. In their model, the BS site activation, antenna type selection, and antenna configuration are encoded as three levels. However, the separation in our model is based on a more inherent difference of the information embedded in a solution, i.e., site activation and user assignment.

6.2.2 Evolution framework

We evolve a population of individuals with adaptive size in the generational mode to approach the optimum. The process starts with randomly generating a population P_0 of a given size. The value of $|P_0|$, i.e., the initial population size, and those of other parameters will be detailed in Section 6.3. Next, each individual's fitness in this initial population is evaluated. Then, the process enters a generational iteration outlined as follows.

1. Randomly pair up individuals of population P_t ($t = 0$ at the start);
2. Crossover each pair of individuals to generate $|P_t|$ offspring;
3. Repair the offspring of previous step;
4. Mutate the offspring;
5. Repair the output of previous step;
6. Evaluate the offspring;
7. Calculate the next population size $|P_{t+1}| = f(|P_t|)$ (see Section 6.2.3);
8. Choose by truncation selection the next population P_{t+1} from the competition pool consist of $|P_t|$ parent and $|P_t|$ offspring individuals according to their fitness;
9. Go to Step 1 if termination condition is not met.

The iterative process stops when the best fitness in the population has remained the same for s (stagnation threshold) individual evaluations. This termination condition will signal if the evolutionary process stagnates. We measure how fast the algorithm leads the process to a possibly local optimum before stagnation by recording the number of individual evaluations elapsed so far.

6.2.3 Adaptive population size

Similar to the adaptive population size approach we introduced in Chapter 4, the rate of evolution measure k_a/k_s ratio is adopted as the adjustment indicator. From one generation to the next, N_a denotes the number of *attempted* nonsynonymous changes and N_s for *attempted* synonymous changes. Specifically, for a crossover, if a valid

offspring alters its fitness from either parent, this crossover is regarded as a nonsynonymous change. A mutation is regarded nonsynonymous if it changes the fitness of an individual. In evolutionary algorithms, not all genetic variations can be favored and accepted by selection. We use M_a and M_s to denote the number of *accepted* nonsynonymous and synonymous changes. Therefore, k_a (k_s resp.) is obtained by dividing the accepted nonsynonymous (synonymous resp.) genetic changes by the attempted nonsynonymous (synonymous resp.) genetic changes.

With the k_a/k_s ratio obtained from each generation, the population size adjustment is performed as follows:

- If $(k_a/k_s)(t) > 1$ (positive selection), we increase the population size proportional to the change in the rate of genetic substitutions such that,

$$|P_{t+1}| = |P_t| \times (1 + |(k_a/k_s)(t) - (k_a/k_s)(t-1)|).$$

- If $(k_a/k_s)(t) = 1$ (neutral selection), we keep the same population size,

$$|P_{t+1}| = |P_t|.$$

- If $(k_a/k_s)(t) < 1$ (negative selection), when $(k_a/k_s)(t)$ is increasing, we increase the population size to suppress further deleterious genetic substitutions, and when $(k_a/k_s)(t)$ is decreasing, we decrease the population size to encourage more genetic substitutions. That is,

$$|P_{t+1}| = |P_t| \times (1 + (k_a/k_s)(t) - (k_a/k_s)(t-1)).$$

Note that, in the truncation selection scheme described in the previous section, the population size of a new generation is at most twice of its previous generation,

and an absolute upper and lower limit of the population size is enforced, as described in Section 6.3.

6.2.4 Evolutionary operations

Crossover

A crossover is applied to two parents, denoted by

$x = \langle x_1, x_2, \dots, x_m \rangle$ and $y = \langle y_1, y_2, \dots, y_m \rangle$, to obtain two children, $x' = \langle x'_1, x'_2, \dots, x'_m \rangle$

and $y' = \langle y'_1, y'_2, \dots, y'_m \rangle$. Crossover is a very important operation in evolutionary algorithm design. The general form of crossover is to exchange certain portions of evolutionary individuals. It is non-trivial to design an efficient crossover operation since it has substantial effects on the performance of an algorithm. Here, we propose a Bi-polar Blend crossover that appropriately incorporates the feature of the network planning problem.

The Bi-polar Blend crossover strives to move the SS assignment from less loaded BSs to more loaded ones so that some will eventually no longer be needed and can be de-activated. Such a crossover is a force to drive the activated BSs towards two extremes, either very heavily or very lightly loaded. Thus, more BSs are expected to be released. To do that, we define that x' inherits the greater load from its parents and y' inherits the less load. Specifically, for each locus i ($1 \leq i \leq m$), we define

$$x'_i = \begin{cases} x_i & \text{if } l(x_i) \geq l(y_i), \\ y_i & \text{otherwise,} \end{cases}$$

and

$$y'_i = \begin{cases} x_i & \text{if } l(x_i) < l(y_i), \\ y_i & \text{otherwise.} \end{cases}$$

Repair heuristic

Note that an individual can become infeasible after the genetic variations. Therefore, we conduct the following greedy repair procedure upon a modified individual, denoted by x . For each $s \in S$, we consider all BSs in x that service it, denoted by \bar{B} . We first remove all overloaded elements in \bar{B} , i.e., load greater than C .

- If $\bar{B} \neq \emptyset$, we keep the most loaded element in \bar{B} and release the rest of \bar{B} .
- Otherwise, i.e., s is not serviced by any BS, we search through all BSs within range to find the *best fit* if any. Here, by best fit we mean, when s is added, the BS that has the least residual capacity. If such a best fit exists, s is added to its load. Note that the identification of such a BS may imply activating previously not-in-service candidate BS site. Otherwise, however, we claim that x cannot be repaired and the current iteration is aborted and the evolutionary process continues with the next iteration.

This repair procedure is equally applicable to the output of both the crossover and mutation operations (next subsection). Note that it also works in such a general trend to drive the activated BSs towards two extremes that more lightly loaded BSs can be released.

Mutation

An individual is subject to a point mutation at the BS activation level. Specifically, we select an activated BS uniformly at random and simply clear its service list. We adopt such a mutation scheme for the following reasons.

- A mutation at the BS activation level, as opposed to the SS assignment level, yields sufficient genetic alteration for solution exploration. A mutation at the SS assignment level, in contrast, would yield a change which is usually too mild.
- Selecting a BS as a unit of mutation confines the changes to one locus of the network. It is, therefore, very well modularized.
- Random selection of an activated BS rather than deterministic, say the least loaded BS, is proved to be less directive and more effective in broadening the exploration space in our preliminary tests.

As this mutation inevitably invalidates the solution, the subsequent repair procedure is also needed.

6.3 Simulation

We are interested in the effectiveness and efficiency of our evolutionary approach. For convenience we will refer to our algorithm as APS-GA (Adaptive Population Size - Genetic Algorithm). Furthermore, to verify the capability of our adaptive population size scheme in improving the algorithm's performance, APS-GA is compared to a FPS-GA (Fixed Population Size - Genetic Algorithm). Computer simulations are designed for these purposes.

6.3.1 Network layouts

Considering that the size and configuration of a network layout may affect the performance of a network planning algorithm, we investigate two scenarios.

Table 6.1: Network configurations.

Parameters	Scenario 1	Scenario 2
SS capacity demand c_s	1	1
BS capacity limit C_B	30	30
SS number $ S $	300	600
BS site number $ B $	30	60
Equivalent coverage range g	0.2	0.15

Table 6.1 presents the network configurations. The bandwidth demands of all SSs are assumed 1 unit and the capacity limit for all BSs is 30, i.e., at most 30 SSs can be connected to a given BS. The deployment area is a 1.0×1.0 2-dimensional space. We consider two network scenarios with 30 (300, resp.) and 60 (600, resp.) BS candidate sites (SSs, resp.). All these sites and nodes are distributed in the space uniformly at random. The channel gains are adjusted so that a BS always has approximately the same number of SSs in range. In all cases, we set the initial population size $|P_0|$ to 200 and the termination stagnation threshold s to 10,000 (evolution is terminated if the best fitness of the population remains unchanged for 10,000 evaluations). Further, we limit the population size to between 100 and 500 when it is varied.

For each scenario, two different layouts are generated, denoted by layout 1.1 and

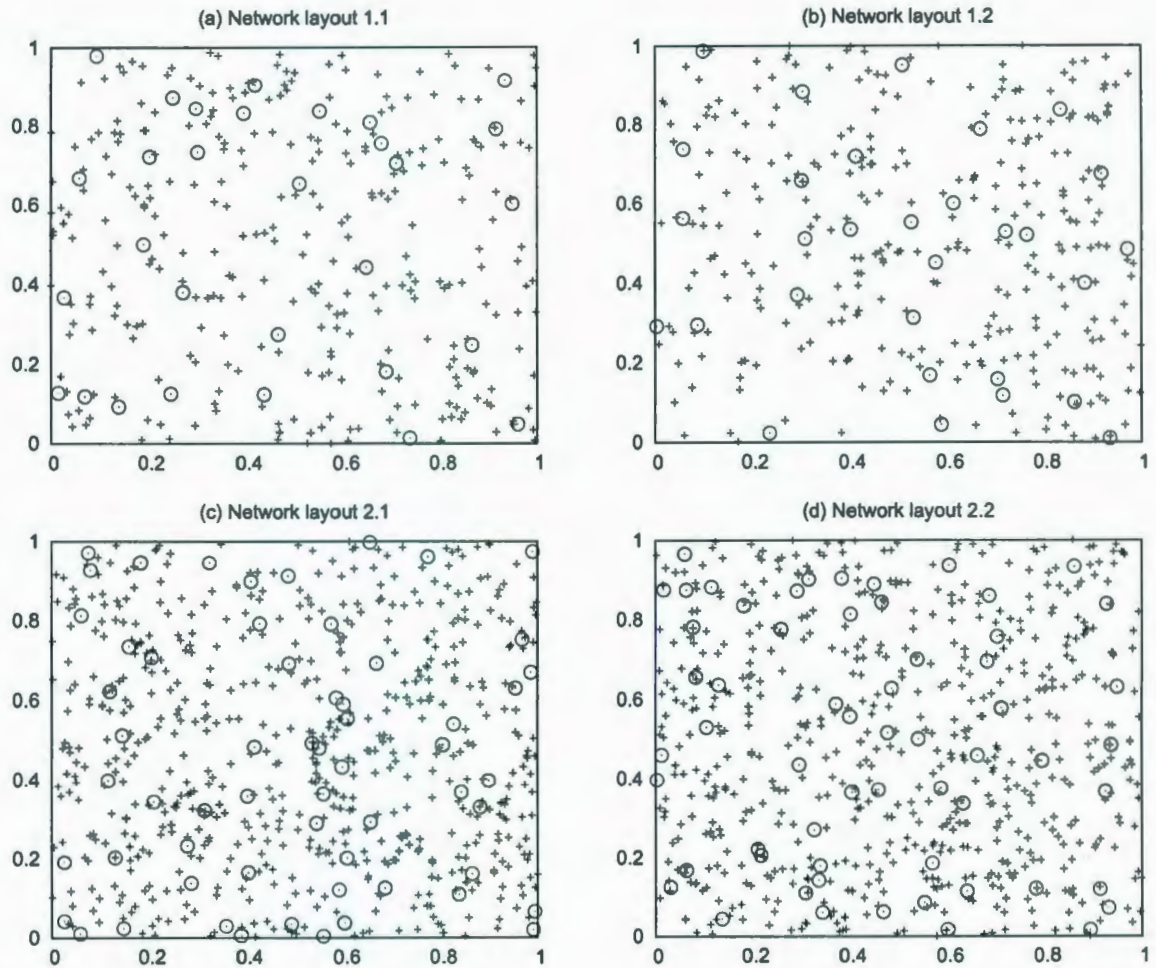


Figure 6.4: Four network layouts in two different scenarios.

1.2 (of scenario 1) and layout 2.1 and 2.2 (of scenario 2). Figure 6.4 shows those four network layouts. In the figures, crosses represent SSs and circles stand for candidate BS locations.

6.3.2 Results

For each network layout, 100 runs of APS-GA are recorded. The fitness of the best solutions (with the minimum usage of BSs) found for four network layouts are 16

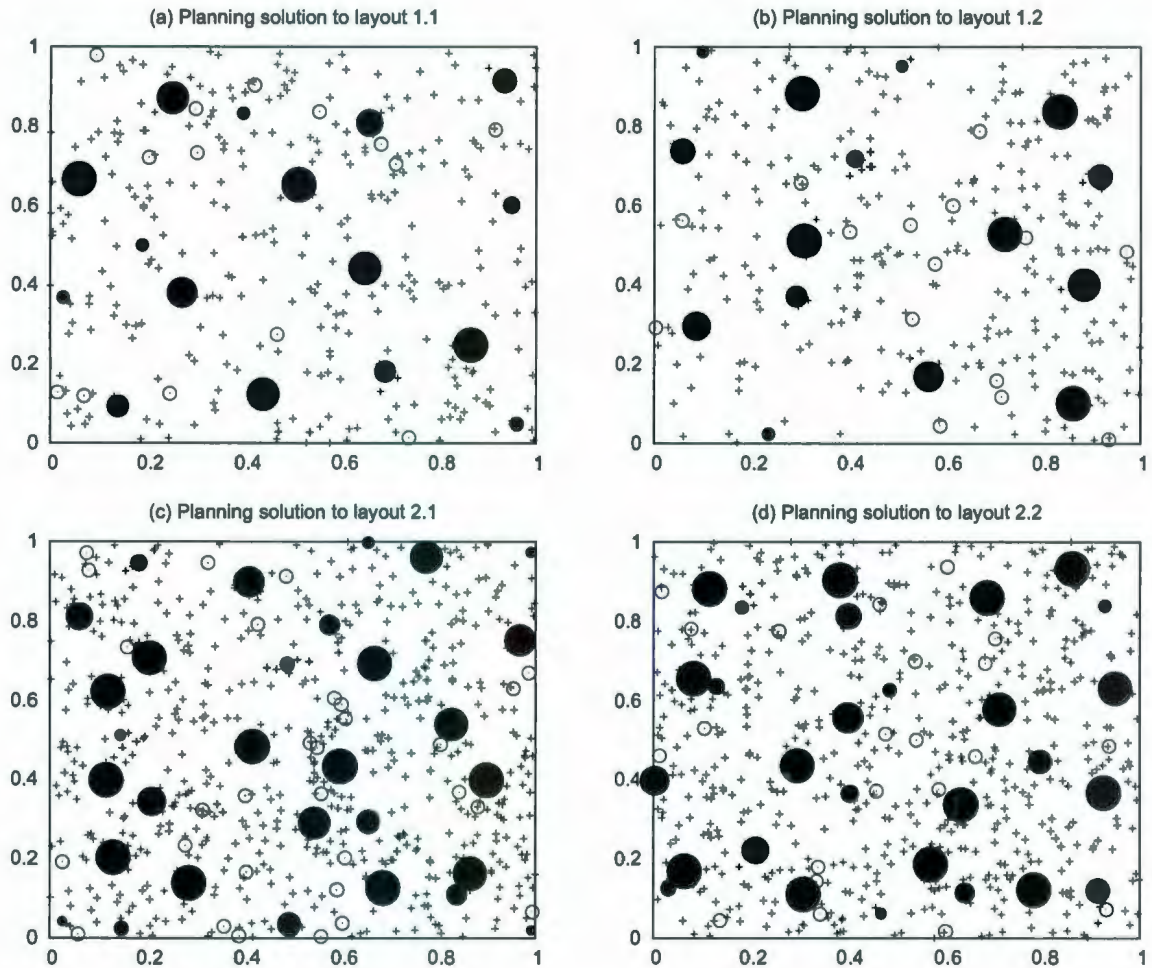


Figure 6.5: Examples of best solutions to the four network layouts.

(layout 1.1), 15 (layout 1.2), 30 (layout 2.1), and 28 (layout 2.2). These show that our method is fairly effective since about half of the candidate BSs can be retired and the average load of active BSs can be as high as 70% of the capacity limit. There also can be more than one best solution for each problem instance.

Figure 6.5 shows four example best solutions generated by APS-GA. In the figures, solid circles represent the BSs in service, and the size of each solid circle indicates the load of the BS it represents. For instance, in Figure 6.5(a), the loads of 16 BSs

in service vary from 5 to 30. We notice that the loads of the BSs do tend to the two extremes, which are expected as a result of our Bi-polar Blend crossover and its corresponding mutation operations. While some BSs are very lightly loaded, they are indispensable to service the entire network.

Table 6.2: Results of APS-GA (average data over 100 runs).

	Layout 1.1	Layout 1.2	Layout 2.1	Layout 2.2
Mean of best fitness	16.0	15.9	30.3	29.1
Mean of evaluations	1862	3450	5082	5315
Median of evaluations	1810	3334	4947	4727
95% confidence interval	[1799,1925]	[3176,3723]	[4805,5360]	[4889,5740]
Mean of population size	310	346	234	250

The statistics of 100 runs of APS-GA are shown in Table 6.2. We collect the mean best fitness achieved at the end of evolution. Recall that the evolution terminates when the best fitness of a population does not improve over 10,000 evaluations. Evaluations before stagnation are recorded as the computational cost for a population to reach its best solution. The means, medians, and the 95% confidence intervals of the number of individual evaluations are shown in the table.

Further, APS-GA is compared to a conventional FPS-GA that has the same operations and parameter configurations as APS-GA. Since the population size fluctuates in APS-GA, we average it during an entire evolutionary process over 100 runs for each problem instance (see the last row in Table 6.2). This average population size will be set as the default and fixed population size for the FPS-GA. Therefore, it is

Table 6.3: Results of FPS-GA (average data over 100 runs).

	Layout 1.1	Layout 1.2	Layout 2.1	Layout 2.2
Mean of best fitness	16.0	16.1	30.2	29.1
Mean of evaluations	2179	3736	5995	5609
Median of evaluations	2170	3459	5732	5250
95% confidence interval	[2157,2201]	[3510,3971]	[5571,6419]	[5199,6019]
Fixed population size	310	346	234	250

possible and fair to compare these two algorithms.

Table 6.3 shows the results from 100 runs of the FPS-GA. It can be observed that the two algorithms perform equally well at achieving best solutions. However, APS-GA is noticeable more efficient since it always incurs smaller computational cost. These results further verify that the evolutionary approach we designed is effective at solving the wireless planning problem. Moreover, the adaptive population size scheme proposed in Chapter 4 has proved again to be able to improve an evolutionary algorithm's performance against the conventional fixed population size algorithms.

6.4 Discussion

The network planning for IEEE 802.16 networks is a constrained combinatorial optimization problem with NP-hardness. A novel evolutionary framework with our adaptive population size scheme is proposed to solve this problem in this chapter.

It is known that standard GA operations are fairly destructive to constrained combinatorial optimization problem and often lead to invalid solutions [144]. To

alleviate this problem, we devise crossover and mutation operations specifically for the network planning problem. The result of our Bi-polar Blend crossover operation is that BSs are either heavily loaded or lightly load. With these two extremes, underutilized BSs can be deleted gradually. Moreover, we use a heuristic repair procedure to maintain the feasibility of a modified individual. Best fit is the strategy we have used, but we understand that they can be an array of different strategies for this. The effectiveness of our evolutionary operations has been verified through computer simulation using our approach to plan the wireless network with four different layouts.

Another motivation is to verify if our adaptive population size scheme can improve an algorithm's performance in such a real world application. Recall that this adaptive population size is implemented according to the rate of a system accepting nonsynonymous to synonymous genetic changes. The central idea of this thesis emphasizes the importance of enhancing neutral search during evolution. Note that in this network planning problem, a considerable number of individuals in a population can have the same fitness. However, they do not necessarily have the same set of BSs in service and associated SSs. The adaptive population size scheme dynamically emphasizes the search with these "neutral" (no fitness improvement) individuals. Since the search space can be enlarged by those neutral explorations, our method is expected to benefit an evolutionary algorithm. This has already been tested in Chapter 4 on a time series prediction problem with GP, and is further confirmed in this chapter by a real world application with GA.

Chapter 7

Concluding Remarks

7.1 Summary

The primary goal of this thesis was to transfer discoveries from Biology to the area of Evolutionary Computation. Evolvability and rate of evolution are the two foci therein. EC, as a heuristic search method, has seen profound developments in both theoretical improvements and application exploration since it was invented based on the general principles from natural evolution. Employing those basic principles from Biology enables EC to be a powerful tool to solve optimization problems from various application areas. It is believed that incorporating new discoveries from modern Biology into our current computation model design can potentially benefit EC to a great extent.

The research on evolvability drives us to look into the fundamental rules in evolution. In this thesis, we consider evolvability as a population property to coordinate various mechanisms and components to enable a system to be evolvable. Neutrality

and variability are the two opposite aspects of evolvability. As they closely cooperate with each other, a system can be resilient to deleterious mutations and sensitive to make adaptive changes at the same time. Such a mechanism explains why an evolvable system can generate adaptive phenotypes from somewhat random genetic changes. When we observe an evolutionary system at two separate levels, genotype and phenotype, we can see that phenotypic changes only take place when a system is adapting to a certain environment. However, the genotypic level generates changes constantly in the background with the majority of them being neutral or nearly neutral. They provide the necessary evolution fuel.

These interesting discoveries are verified in GP systems, a branch of the EC family. This thesis contributes to the understanding of general evolution principles since computation systems are artificial, and thus easy to control and track. Furthermore, with better understanding of evolvability, EC researchers can concentrate on enhancing this evolvability to improve computation models.

We also present an example of employing some approaches and theories from Biology to improve the performance of an EC algorithm. A rate of evolution measurement k_a/k_s ratio is adopted from molecular biology to GP. Biologists employ the metrics based on genetic activities to quantify the rate of evolution on protein-coded gene sequences, due to the infeasibility of defining fitness quantitatively in natural organisms. Although we have explicitly defined fitness in EC, this k_a/k_s ratio suggests to us a different channel to observe an evolutionary process at a deeper level of evolution dynamics. As a potential to evolve, evolvability is also a “second order” effect of fitness. This ratio also provides a very useful tool to study evolvability.

This thesis formulates the calculation of this k_a/k_s ratio in EC. Simulation shows

the effectiveness of this measurement in quantifying rate of evolution in GP. Further through the investigations on major configuration parameters, a non-monotonic relationship between population size and rate of evolution is reported. The *Nearly Neutral Theory* from population genetics is reviewed to explain the role of population size in rate of evolution. Inspired by this theory, we propose an adaptive population size approach which adjusts the size of a population dynamically during evolution according to the rate of evolution measurement k_a/k_s ratio. This population size adjustment practically encourages neutrality exploration during stagnant periods of evolution. Experimental studies on a GP system endorse our observations as this adaptive population size approach can effectively improve its search performance compared to fixed-size populations.

The rate of evolution measure k_a/k_s ratio and the adaptive population size approach are further incorporated into a Genetic Algorithm (GA), another branch of the EC family, to solve a real world application problem, the wireless network planning. Specific individual representation and evolutionary operations are designed for this particular problem, as well as employing the core idea of this thesis in action, i.e., enhancing neutral search in evolution by varying the size of a population. Simulation results again verify the effectiveness of our methods to improve the performance of an EC system in the context of a real world application.

7.2 Future research

The ideas and methods proposed in this thesis can be further refined and explored as follows:

- The rate of evolution measure k_b/k_s ratio in EC has shown to be very effective to reflect the rate of evolution, to design an adaptive population size scheme, and to investigate evolvability. Since this is a widely used tool in the biological literature already, further application of this measure in EC seems promising.
- A formal and well accepted definition of evolvability and its quantification method have not been achieved at this point in the literature. This is a quite open research area and there are many possible directions in the near future. For instance, it has been pointed out that the organism-environment interaction is crucial for investigating evolvability. Therefore, analysis of dynamic environments with more complex changing patterns and intensity will be interesting.
- As better understanding of evolvability is attained, we hope to improve the evolvability of computational evolution systems. Among the reviewed new biological developments related to evolvability in this thesis, epigenetic mechanisms are anticipated to play an important role in increasing the evolvability of EC algorithms. The reason is that the epigenetic regulation reveals a considerably complex and intelligent interactive system of gene expression in living organisms. Such a complex system possesses a large amount of feedback information, from both the environmental challenges and the intrinsic interactions among various components inside an organism, in order to supervise its gene expression. These mechanisms, therefore, make living organisms very resilient and adaptive in evolution. However, this system is quite distinguishing from the unilateral control flow of common EC models, which is also the reason that it can inspire future innovation in algorithm design.

- Neutral and nearly neutral mutations that have little effect on phenotypes have received increasing interest in Biology. The importance of neutral search has also been revealed in EC that it in fact explores larger search spaces. This observation challenges the traditional belief that redundancy is hardly useful beyond providing certain protection against deleterious genetic changes. Therefore, neutrality is another promising future research direction. It will be very interesting to monitor the process of individuals searching and moving in neutral networks. This may be a better explanation for why neutral search can benefit an EC algorithm.

Recall that we stated at the beginning of this thesis that EC is an optimization approach inspired by mechanisms of natural evolution. It is important to incorporate new discoveries from natural evolution. The notions and ideas brought in this thesis focus on evolvability and rate of evolution. However, there are many other aspects that need to be explored in this light, which might be rewarding to algorithm design.

From a methodological point of view, although this thesis show-cases an example that applying new discoveries from Biology is beneficial to EC, we should be aware that there are always limitations both in these discoveries themselves and their applications to EC. On one hand, we know that the study of biological systems has developed profoundly during the past decades. However, the core mechanisms of evolution in living organisms are still far from being clearly understood. We are in the century that new discoveries and new ideas come up faster than ever before. This is both challenging and exciting for interdisciplinary research. For instance, related to the central idea of this thesis, current research by Kudla et al. [95] suggests that

mutations on synonymous sites of a gene sequence can influence the level of gene expression even though they do not alter the encoded protein. Some silent mutations can affect the transcribed messenger RNA (mRNA) folding and its properties, which play an important role in shaping expression levels of genes. Therefore, those synonymous mutations somehow affect gene expression. This finding challenges the conventional definition of "synonymous" mutations which were regarded not functional on the phenotype. In EC, current computational models barely follow a rough analogy to the much more complex living systems. Particularly, the role of mRNA in most EC algorithms is overlooked in most cases. Yet, this shows great potential for incorporating new biological discoveries to design more complex and intelligent algorithms. Therefore, close investigation of the state-of-the-art in biological literature is crucial for EC research. On the other hand, not all biological notions and principles are suitable for EC models since these two systems are substantially different from each other. Therefore, significant endeavors are still needed to study the similarities and differences between them. More caution would be taken when transferring ideas and principles from Biology to EC.

Bibliography

- [1] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J. D. Watson. *Molecular Biology of the Cell (4th edition)*. Garland, New York, NY, USA, 2002.
- [2] C. D. Allis, T. Jenuwein, and D. Reinberg, editors. *Epigenetics*. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, USA, 2007.
- [3] C. D. Allis, T. Jenuwein, and D. Reinberg. Overview and concepts. In *Epigenetics*, pages 23–61. Cold Spring Harbor Laboratory Press, Cold Spring Harbor, NY, USA, 2007.
- [4] O. Alp, E. Erkut, and Z. Drezner. An efficient genetic algorithm for the p-median problem. *Annals of Operations Research*, 122(1-4):21–42, September 2003.
- [5] L. Altenberg. The evolution of evolvability in genetic programming. In *Advances in Genetic Programming*, pages 47–74. MIT Press, Cambridge, MA, USA, 1994.
- [6] L. Altenberg. The schema theorem and price's theorem. In *Foundations of Genetic Algorithms 3*, pages 23–49. MIT Press, 1995.

- [7] A. Amaldi, A. Capone, and F. Malucelli. Planning UMTS base station location: optimization models with power control and algorithms. *IEEE Transactions on Wireless Communications*, 2(5):939–952, September 2003.
- [8] J. Arabas, Z. Michalewicz, and J. Mulawka. GAVaPS – a genetic algorithm with varying population size. In *Proceedings of the 1st IEEE Conference on Evolutionary Computation*, pages 73–78. IEEE Press, 1994.
- [9] S. Asthana, W. S. Noble, G. Kryukov, C. E. Grant, S. Sunyaev, and J. A. Stamatoyannopoulos. Widely distributed noncoding purifying selection in the human genome. *Proceedings of the National Academy of Sciences*, 104(30):12410–12415, 2007.
- [10] T. Back, A. E. Eiben, and N. A. L. van der Vaart. An empirical study on GAs “without parameters”. In *Proceedings of the 6th International Conference on Parallel Problem Solving from Nature (PPSN VI)*, volume 1917 of *Lecture Notes in Computer Science*, pages 315–324. Springer, 2000.
- [11] W. Banzhaf. Artificial regulatory networks and genetic programming. In *Genetic Programming Theory and Practice*, pages 43–62. Kluwer Academic, 2003.
- [12] W. Banzhaf. On the dynamics of an artificial regulatory networks. In *Advances in Artificial Life – Proceedings of the 7th European Conference on Artificial Life*, volume 2801 of *Lecture Notes in Artificial Intelligence*, pages 217–227. Springer-Verlag, 2003.
- [13] W. Banzhaf, G. Beslon, S. Christensen, J. A. Foster, F. Kepes, V. Lefort, J. F. Miller, M. Radman, and J. J. Ramsden. From artificial evolution to computa-

- tional evolution: A research agenda. *Nature Reviews Genetics*, 7(9):729–735, 2006.
- [14] W. Banzhaf and P. Kuo. Network motifs in natural and artificial transcriptional regulatory networks. *Journal of Biological Physics and Chemistry*, 4(2):85–92, 2004.
- [15] W. Banzhaf and A. Leier. Evolution on neutral networks in genetic programming. In *Genetic Programming Theory and Practice III*, pages 207–221. Springer, 2006.
- [16] W. Banzhaf, P. Nordin, R. E. Keller, and F. D. Francone. *Genetic Programming – An Introduction: On the Automatic Evolution of Computer Programs and its Applications*. Morgan Kaufmann, San Francisco, CA, USA, 1998.
- [17] S. B. Baylin. DNA hypermethylation in tumorigenesis. *Trends in Genetics*, 16(4):168–174, 2000.
- [18] M. Beckerman. *Molecular and Cellular Signaling*. Springer, New York, NY, USA, 2005.
- [19] M. A. Bedau. Quantifying the extent and intensity of adaptive evolution. In *GECCO'99: Proceedings of the 1999 Genetic and Evolution Computation Conference Workshop Program*, pages 34–37. Morgan Kaufmann, 1999.
- [20] M. A. Bedau and N. H. Packard. Measurement of evolutionary activity, teleology, and life. In *Artificial Life II*, pages 431–461. Addison-Wesley, Redwood City, CA, USA, 1992.

- [21] M. A. Bedau and N. H. Packard. Evolution of evolvability via adaptation of mutation rates. *BioSystems*, 69(2):143–162, 2003.
- [22] M. A. Bedau and R. Seymour. Adaptation of mutation rates in a simple model of evolution. In *Complex Systems: Mechanism of Adaptation*, pages 179–183. IOS Press, Amsterdam, The Netherlands, 1994.
- [23] T. V. Belle and D. H. Ackley. Code factoring and the evolution of evolvability. In *GECCO '02: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1383–1390. ACM, 2002.
- [24] T. Blickle and L. Thiele. A comparison of selection schemes used in evolutionary algorithms. *Evolutionary Computation*, 4(4):361–394, 1997.
- [25] J. T. Bonner. *The evolution of complexity by means of natural selection*. Princeton University Press, Princeton, NJ, USA, 1988.
- [26] R. Calabretta, S. Nolfi, and G. P. Wagner. Duplication of modules facilitates the evolution of functional specialization. *Artificial Life*, 6(1):69–84, 2000.
- [27] S. Carroll, J. Grenier, and S. Weatherbee. *From DNA to Diversity*. Blackwell Publishing, 2005.
- [28] A. Chavoya and Y. Duthen. Evolving an artificial regulatory network for 2D cell patterning. In *ALIFE'07: IEEE Symposium on Artificial Life 2007*, pages 47–53. IEEE Press, 2007.
- [29] J. Cohen. Bioinformatics—an introduction for computer scientists. *ACM Computing Surveys*, 36(2):122–158, 2004.

- [30] S. Collins, J. de Meaux, and C. Acquisti. Adaptive walks toward a moving optimum. *Genetics*, 176(2):1089–1099, 2007.
- [31] M. Conrad. The geometry of evolution. *BioSystems*, 24(11):61–81, 1990.
- [32] E. S. Correa, M. T. A. Steiner, A. A. Freitas, and C. Carnieri. A genetic algorithm for solving a capacitated p-median problem. *Numerical Algorithms*, 35(2-4):373–388, April 2004.
- [33] J. F. Crow and M. Kimura. *An Introduction to Population Genetics Theory*. Harper and Row Publishers, 1970.
- [34] C. Darwin. *On the Origin of Species by means of natural selection*. John Murray, London, 1859.
- [35] R. Dawkins. The evolution of evolvability. In *Artificial Life: The Quest for a New Creation*, pages 201–220. Addison-Wesley, Reading, MA, USA, 1989.
- [36] J. A. G. M. de Visser, J. Hermisson, G. P. Wagner, L. A. Meyers, H. Bagheri-Chaichian, J. L. Blanchard, L. Chao, J. M. Cheverud, S. F. Elena, W. Fontana, G. Gibson, T. F. Hansen, D. Krakauer, R. C. Lewontin, C. Ofria, S. H. Rice, G. von Dassow, A. Wagner, and M. C. Whitlock. Perspective: Evolution and detection of genetic robustness. *Evolution*, 57(9):1959–1972, 2003.
- [37] R. M. Downing. On population size and neutrality: Facilitating the evolution of evolvability. In *Proceedings of the 10th European Conference on Genetic Programming (EuroGP 2007)*, volume 4445 of *Lecture Notes in Computer Science*, pages 181–192. Springer, 2007.

- [38] D. Dykhuizen and D. L. Hartl. Selective neutrality of 6PGD Allozymes in *E. Coli* and the effects of genetic background. *Genetics*, 96(4):801–817, 1980.
- [39] D. J. Earl and M. W. Deem. Evolvability is a selectable trait. *Proceedings of the National Academy of Sciences*, 101(32):11531–11536, 2004.
- [40] M. Ebner, M. Shackleton, and R. Shipman. How neutral networks influence evolvability. *Complexity*, 7(2):19–33, 2002.
- [41] A. E. Eiben. Multiparent recombination in evolutionary computing. In *Advances in evolutionary computing: theory and applications*, pages 175–192. Springer-Verlag New York, Inc., New York, NY, USA, 2003.
- [42] A. E. Eiben, E. Marchiori, and V. A. Valkó. Evolutionary algorithms with on-the-fly population size adjustment. In *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, pages 41–50. Springer, 2004.
- [43] C. Eklund, R. B. Marks, K. L. Stanwood, and S. Wang. IEEE standard 802.16: a technical overview of the WirelessMAN air interface for broadband wireless access. *IEEE Communications Magazine*, 41(6):98–107, June 2002.
- [44] N. Eldredge and S. J. Gould. Punctuated Equilibria: An alternative to phyletic gradualism. In *Models in Paleobiology*, pages 82–115. Freeman Cooper and Co., San Francisco, CA, USA, 1972.

- [45] E. Falkenauer and A. Delchambre. A genetic algorithm for bin packing and line balancing. In *Proceedings of the IEEE 1992 International Conference on Robotics and Automation*, pages 1186–1192, 1992.
- [46] C. Fernandes and A. Rosa. Self-regulated population size in evolutionary algorithms. In *Proceedings of the 9th International Conference on Parallel Problem Solving from Nature (PPSN IX)*, volume 4193 of *Lecture Notes in Computer Science*, pages 920–929. Springer, 2006.
- [47] R. A. Fisher. *Genetical Theory of Natural Selection*. Clarendon, Oxford, UK, 1930.
- [48] D. B. Fogel. *System Identification through Simulated Evolution: A Machine Learning Approach to Modeling*. Ginn Press, Needham Heights, MA, USA, 1991.
- [49] L. J. Fogel, A. J. Owens, and M. J. Walsh. *Artificial Intelligence through Simulated Evolution*. John Wiley and Sons, New York, NY, USA, 1966.
- [50] J. W. Fondon III and H. R. Garner. Molecular origins of rapid and continuous morphological evolution. *Proceedings of the National Academy of Sciences*, 101(52):18058–18063, 2004.
- [51] G. Forgacs and S. Newman. *Biological Physics of the Developing Embryo*. Cambridge University Press, 2005.

- [52] D. Fredman, S. J. White, S. Potter, E. E. Eichler, J. T. D. Dunnen, and A. J. Brookes. Complex SNP-related sequence variation in segmental genome duplications. *Nature Genetics*, 36(8):861–866, 2004.
- [53] J. L. Freeman, G. H. Perry, L. Feuk, R. Redon, S. A. McCarroll, D. M. Altshuler, H. Aburatani, K. W. Jones, C. Tyler-Smith, M. E. Hurles, N. P. Carter, S. W. Scherer, and C. Lee. Copy number variation: New insights in genome diversity. *Genome Research*, 16(8):949–961, 2006.
- [54] M. C. Frith, M. Pheasant, and J. S. Mattick. The amazing complexity of the human transcriptome. *European Journal of Human Genetics*, 13(8):894–897, 2005.
- [55] D. Ghosh, A. Gupta, and P. Mohapatra. Scheduling in multihop WiMAX networks. *ACM Mobile Computing and Communications Review*, 12(2):1–11, April 2007.
- [56] J. H. Gillespie. The role of population size in molecular evolution. *Theoretical Population Biology*, 55(2):145–156, 1999.
- [57] J. H. Gillespie. *Population Genetics: a Concise Guide*. Johns Hopkins University Press, 2004.
- [58] D. E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Boston, MA, USA, 1989.

- [59] D. E. Goldberg. Sizing populations for serial and parallel genetic algorithms. In *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 70–79. Morgan Kaufmann, 1989.
- [60] D. E. Goldberg and K. Deb. A comparative analysis of selection schemes used in genetic algorithms. In *Foundations of Genetic Algorithms 1*, pages 69–93. Morgan Kaufmann, 1991.
- [61] D. E. Goldberg, K. Deb, and J. H. Clark. Genetic algorithms, noise, and the sizing of populations. *Complex Systems*, 6(4):333–362, 1992.
- [62] D. E. Goldberg, B. Korb, and K. Deb. Messy genetic algorithms: Motivation, analysis, and first results. *Complex Systems*, 3(5):493–530, 1989.
- [63] S. Gould. *The Structure of Evolutionary Theory*. Harvard University Press, 2002.
- [64] S. I. S. Grewal and D. Moazed. Heterochromatin and epigenetic control of gene expression. *Science*, 301(5634):798–802, 2003.
- [65] F. Gruau. Genetic micro programming of neural networks. In *Advances in genetic programming*, pages 495–518. MIT Press, Cambridge, MA, USA, 1994.
- [66] X. Gu, Z. Zhang, and W. Huang. Rapid evolution of expression and regulatory divergences after yeast gene duplication. *Proceedings of the National Academy of Sciences*, 102(3):707–712, 2005.

- [67] G. R. Harik and F. G. Lobo. A parameter-less genetic algorithm. In *GECCO'99: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 258–267. Morgan Kaufmann, 1999.
- [68] P. M. Harrison, H. Hegyi, S. Balasubramanian, N. M. Luscombe, P. Bertone, N. Echols, T. Johnson, and M. Gerstein. Molecular fossils in the human genome: Identification and analysis of the pseudogenes in chromosomes 21 and 22. *Genome Research*, 12(2):272–280, 2002.
- [69] I. Harvey. The SAGA cross: the mechanics of recombination for species with variable length genotypes. In *Proceedings of the 2ed International Conference on Parallel Problem Solving from Nature (PPSN II)*, pages 269–278. North-Holland, 1992.
- [70] J. Hawks, E. T. Wang, G. M. Cochran, H. C. Harpending, and R. K. Moyzis. Recent acceleration of human adaptive evolution. *Proceedings of the National Academy of Sciences*, 104(52):20753–20758, 2007.
- [71] H. Hendrickson, E. S. Slechta, U. Bergthorsson, D. I. Andersson, and J. R. Roth. Amplification-mutagenesis: Evidence that “directed” adaptive mutation and general hypermutability result from growth with a selected gene amplification. *Proceedings of the National Academy of Sciences*, 99(4):2164–2169, 2002.
- [72] A. P. Hendry. The power of natural selection. *Nature*, 433(7027):694–695, 2005.
- [73] J. H. Holland. *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI, USA, 1975.

- [74] T. Hu and W. Banzhaf. Measuring rate of evolution in genetic programming using amino acid to synonymous substitution ratio k_a/k_s . In *GECCO '08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pages 1337–1338. ACM, 2008.
- [75] T. Hu and W. Banzhaf. Nonsynonymous to synonymous substitution ratio k_a/k_s : Measurement for rate of evolution in evolutionary computation. In *Proceedings of the 10th International Conference on Parallel Problem Solving from Nature (PPSN X)*, volume 5199 of *Lecture Notes in Computer Science*, pages 448–457. Springer, 2008.
- [76] T. Hu and W. Banzhaf. Neutrality and variability: Two sides of evolvability in linear genetic programming. In *GECCO '09: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, pages 963–970. ACM, 2009.
- [77] T. Hu and W. Banzhaf. The role of population size in rate of evolution in genetic programming. In *Proceedings of the 12th European Conference on Genetic Programming (EuroGP)*, volume 5481 of *Lecture Notes in Computer Science*, pages 85–96. Springer, 2009.
- [78] T. Hu, Y. P. Chen, W. Banzhaf, and R. Benkoczi. An evolutionary approach to planning ieee 802.16 networks. In *GECCO '09: Proceedings of the 11th Annual Conference on Genetic and Evolutionary Computation*, pages 1929–1930. ACM, 2009.

- [79] A. J. Iafrate, L. Feuk, M. N. Rivera, M. L. Listewnik, P. K. Donahoe, Y. Qi, and S. W. S. C. Lee. Detection of large-scale variation in the human genome. *Nature Genetics*, 36(9):949–951, 2004.
- [80] E. Jablonka and M. J. Lamb. *Evolution in Four Dimensions: Genetic, Epigenetic, Behavioral, and Symbolic Variation in the History of Life*. MIT Press, Cambridge, MA, USA, 2005.
- [81] F. Jacob. Evolution and tinkering. *Science*, 196(4295):1161–1166, 1977.
- [82] R. Jaenisch and A. Bird. Epigenetic regulation of gene expression: How the genome integrates intrinsic and environmental signals. *Nature Genetics*, 33:245–254, 2003.
- [83] Y. Jin. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing - A Fusion of Foundations, Methodologies and Applications*, 9(1):3–12, 2005.
- [84] N. Kashtan and U. Alon. Spontaneous evolution of modularity and network motifs. *Proceedings of the National Academy of Sciences*, 102(39):13773–13778, 2005.
- [85] N. Kashtan, E. Noor, and U. Alon. Varying environments can speed up evolution. *Proceedings of the National Academy of Sciences*, 104(34):13713–13716, 2007.
- [86] S. T. Kazadi, D. Lee, R. Modi, J. Sy, and W. Lue. Levels of compartmentalization in artificial evolution. In *GECCO'00: Proceedings of the Genetic*

- and Evolutionary Computation Conference*, pages 841–848. Morgan Kaufmann, 2000.
- [87] M. C. King and A. C. Wilson. Evolution at two levels in humans and chimpanzees. *Science*, 188(4184):107–116, 1975.
- [88] M. W. Kirschner and J. C. Gerhart. Evolvability. *Proceedings of the National Academy of Sciences*, 95(15):8420–8427, 1998.
- [89] M. W. Kirschner and J. C. Gerhart. *The Plausibility of Life*. Yale University Press, New Haven, CT, USA, 2005.
- [90] A. Kong, D. F. Gudbjartsson, J. Sainz, G. M. Jonsdottir, S. A. Gudjonsson, B. Richardsson, S. Sigurdardottir, J. Barnard, B. Hallbeck, G. Masson, A. Shlien, S. T. Palsson, M. L. Frigge, T. E. Thorgeirsson, J. R. Gulcher, and K. Stefansson. A high-resolution recombination map of the human genome. *Nature Genetics*, 31(3):241–247, 2002.
- [91] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, Cambridge, MA, USA, 1992.
- [92] J. R. Koza. *Genetic programming II: automatic discovery of reusable programs*. MIT Press, Cambridge, MA, 1994.
- [93] J. R. Koza. Gene duplication to enable genetic programming to concurrently evolve both the architecture and work-performing steps of a computer program. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 695–717. Morgan Kaufmann, 1995.

- [94] D. C. Krakauer and J. B. Plotkin. Redundancy, antiredundancy, and the robustness of genomes. *Proceedings of the National Academy of Sciences*, 99(3):1405–1409, 2002.
- [95] G. Kudla, A. W. Murray, D. Tollervey, and J. B. Plotkin. Coding-sequence determinants of gene expression in *Escherichia coli*. *Science*, 324(5924):255–258, 2009.
- [96] E. Kugelberg, E. Kofoed, A. B. Reams, D. I. Andersson, and J. R. Roth. Multiple pathways of selected gene amplification during adaptive mutation. *Proceedings of the National Academy of Sciences*, 103(46):17319–17324, 2006.
- [97] P. Kuo, W. Banzhaf, and A. Leier. Network topology and the evolution of dynamics in an artificial genetic regulatory network model created by whole genome duplication and divergence. *Biosystems*, 85(3):177–200, 2006.
- [98] W. B. Langdon. Size fair and homologous tree genetic programming crossovers. *Genetic Programming and Evolvable Machines*, 1(1):95–119, 2000.
- [99] W. B. Langdon and W. Banzhaf. Repeated patterns in tree genetic programming. In M. Keijzer, A. Tettamanzi, P. Collet, M. Tomassini, and J. van Hemert, editors, *Proceedings of the 8th European Conference on Genetic Programming*, pages 190–202, Lausanne, Switzerland, 2005.
- [100] W. B. Langdon, T. Soule, R. Poli, and J. A. Foster. The evolution of size and shape. In *Advances in Genetic Programming: Volume 3*, pages 163–190. MIT Press, Cambridge, MA, USA, 1999.

- [101] J. E. G. Leigh. Levels of selection, potential conflicts, and their resolution: The role of the “common good”. In *Levels of Selection in Evolution*, pages 15–30. Princeton University Press, Princeton, NJ, USA, 1999.
- [102] B. Li, Y. Qin, C. P. Low, and C. L. Gwee. A survey on Mobile WiMAX. *IEEE Communications Magazine*, 46(12):70–75, December 2007.
- [103] K. F. Liem. Key evolutionary innovations, differential diversity, and symecomorphosis. In *Evolutionary Innovations*, pages 147–170. University of Chicago Press, Chicago, IL, USA, 1990.
- [104] P. Lin, H. Ngo, C. Qiao, X. Wang, T. Wang, and D. Qian. Minimum cost wireless broadband overlay network planning. In *IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks (WoWMoM)*, pages 228–236, Buffalo, NY, June 2006.
- [105] F. G. Lobo and C. F. Lima. A review of adaptive population sizing schemes in genetic algorithms. In *GECCO’05: Proceedings of the 2005 workshops on Genetic and Evolutionary Computation*, pages 228–234. ACM, 2005.
- [106] M. Z. Ludwig, N. H. Patel, and M. Kreitman. Functional analysis of eve stripe 2 enhancer evolution in *Drosophila*: Rules governing conservation and change. *Development*, 125(5):949–958, 1998.
- [107] S. Luke. Code growth is not caused by introns. In *GECCO’00: Late Breaking Papers at the 2000 Genetic and Evolutionary Computation Conference*, pages 228–235. Morgan Kaufmann, 2000.

- [108] S. Luke and L. Panait. A comparison of bloat control methods for genetic programming. *Evolutionary Computation*, 14(3):309–334, 2006.
- [109] S. C. Manrubia and C. Briones. Modular evolution and increase of functional complexity in replicating RNA molecules. *RNA*, 13(1):97–107, 2007.
- [110] P. Marrow, M. Heath, and I. I. Re. Evolvability: Evolution, Computation, and Biology. In *GECCO'99: Proceedings of the 1999 Genetic and Evolutionary Computation Conference Workshop on Evolvability*, pages 30–33. Morgan Kaufmann, 1999.
- [111] H. D. Marshall, M. W. Coulson, and S. M. Carr. Near neutrality, rate heterogeneity, and linkage govern mitochondrial genome evolution in atlantic cod (*Gadus morhua*) and other gadine fish. *Molecular Biology and Evolution*, 26(3):579–589, 2009.
- [112] J. S. Mattick. Non-coding RNAs: the architects of eukaryotic complexity. *EMBO Reports*, 2(11):986–991, 2001.
- [113] J. S. Mattick. Challenging the dogma: The hidden layer of non-protein-coding RNAs in complex organisms. *BioEssays*, 25(10):930–939, 2003.
- [114] J. S. Mattick. RNA regulation: a new genetics? *Nature Reviews Genetics*, 5(4):316–323, 2004.
- [115] C. Mattiussi and D. Floreano. Analog genetic encoding for the evolution of circuits and networks. *IEEE Transactions on Evolutionary Computation*, 11(5):596–607, 2007.

- [116] E. Mayr and W. Provine, editors. *The Evolutionary Synthesis*. Harvard University Press, 1998, 2nd edition.
- [117] H. Meunier, E. Ghazali Talbi, and P. Reininger. A multiobjective genetic algorithm for radio network optimization. In *Proceedings of the 2000 Congress on Evolutionary Computation (CEC 2000)*, pages 317–324. IEEE, 2000.
- [118] L. A. Meyers, F. D. Ancel, and M. Lachmann. Evolution of genetic potential. *PLoS Computational Biology*, 1(3):0236–0243, 2005.
- [119] T. Miyata and T. Yasunaga. Molecular evolution of mRNA: A method for estimating evolutionary rates of synonymous and amino acid substitutions from homologous nucleotide sequences and its application. *Journal of Molecular Evolution*, 16(1):23–36, 1980.
- [120] S. V. Muse and B. S. Gaut. A codon-based model of nucleotide substitution for protein-coding DNA sequences. *Molecular Biology and Evolution*, 11(5):725–736, September 1994.
- [121] C. L. Nehaniv. Measuring evolvability as the rate of complexity increase. In *Proceedings of the Artificial Life VII Workshop*, pages 55–57. MIT Press, 2000.
- [122] C. L. Nehaniv. Self-replication, evolvability and asynchronicity in stochastic worlds. In *Stochastic Algorithms: Foundations and Applications*, pages 126–169. Springer, 2005.
- [123] M. Nei. Gene duplication and nucleotide substitution in evolution. *Nature*, 221(5175):40–42, 1969.

- [124] M. Nei. Selectionism and neutralism in molecular evolution. *Molecular Biology and Evolution*, 22(12):2318–2342, 2005.
- [125] M. Nei and A. P. Rooney. Concerted and birth-and-death evolution of multigene families. *Annual Review of Genetics*, 22(15):121–152, 2005.
- [126] S. R. Neves and R. Iyengar. Modeling of signaling networks. *BioEssays*, 24(12):1110–1117, 2002.
- [127] P. Nordin, W. Banzhaf, and F. Francone. Efficient evolution of machine code for cisc architectures using blocks and homologous crossover. In *Advances in Genetic Programming: Volume 3*, pages 275–299. MIT Press, Cambridge, MA, USA, 1999.
- [128] P. Nordin, F. Francone, and W. Banzhaf. Explicitly defined introns and destructive crossover in genetic programming. In *Advances in Genetic Programming: Volume 2*, pages 111–134. MIT Press, Cambridge, MA, USA, 1996.
- [129] H. Oakley. Two scientific applications of genetic programming: stack filters and non-linear equation fitting to chaotic data. In *Advances in genetic programming*, pages 369–389. MIT Press, 1994.
- [130] S. Ohno. *Evolution by Gene Duplication*. Springer-Verlag, New York, NY, USA, 1970.
- [131] T. Ohta. Population size and rate of evolution. *Journal of Molecular Evolution*, 1(4):305–314, 1972.

- [132] T. Ohta. The nearly neutral theory of molecular evolution. *Annual Review of Ecology and Systematics*, 23:263–286, 1992.
- [133] H. A. Orr. The population genetics of adaptation: the distribution of factors fixed during adaptive evolution. *Evolution*, 52(4):935–949, 1998.
- [134] H. A. Orr. Fitness and its role in evolutionary genetics. *Nature Review Genetics*, 10:531–539, 2009.
- [135] S. Palumbi. *The Evolution Explosion*. Norton, New York, 2002.
- [136] M. Parter, N. Kashtan, and U. Alon. Facilitated variation: How evolution learns from past environments to generalize to new environments. *PLoS Computational Biology*, 4(11):e1000206, 2008.
- [137] J. W. Pepper. The evolution of evolvability in genetic linkage patterns. *BioSystems*, 69(2):115–126, 2003.
- [138] R. Poli and N. F. McPhee. Exact schema theory for GP and variable-length GAs with homologous crossover. In *GECCO'01: Proceedings of the Genetic and Evolutionary Computation Conference*, pages 104–111. Morgan Kaufmann, 2001.
- [139] R. Poli, N. F. McPhee, and L. Vanneschi. The impact of population size on code growth in GP: analysis and empirical validation. In *GECCO '08: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, pages 1275–1282. ACM, 2008.

- [140] D. Posada, K. A. Crandall, and E. C. Holmes. Recombination in evolutionary genomics. *Annual Review of Genetics*, 36:75–97, 2002.
- [141] R. A. Raff. *The Shape of Life: Genes, Development, and the Evolution of Animal Form*. University of Chicago Press, Chicago, IL, USA, 1996.
- [142] M. J. Raven and M. A. Bedau. General framework for evolutionary activity. In *Proceedings of the Seventh European Conference on Artificial Life*, pages 676–685. Springer-Verlag, 2003.
- [143] I. Rechenberg. *Evolution Strategy*. Frommann-Holzboog, Stuttgart, Germany, 1973.
- [144] C. Reeves. Hybrid genetic algorithms for bin-packing and related problems. *Annals of Operations Research*, 63:371–396, 1996.
- [145] J. Reisinger and R. Miikkulainen. Selecting for evolvable representations. In *GECCO'06: Proceedings of the 2006 Genetic and Evolutionary Computation Conference*, pages 1297–1304. ACM, 2006.
- [146] J. Reisinger, K. O. Stanley, and R. Miikkulainen. Towards an empirical measure of evolvability. In *GECCO'05: Proceedings of the 2005 Genetic and Evolutionary Computation Conference Workshop Program*, pages 257–264. ACM, 2005.
- [147] C. S. Retsef Levi, David B. Shmoys. LP-based approximation algorithms for capacitated facility location. In *10th International Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 206–218, New York, NY, June 2004.

- [148] F. Roderiguez-trelles, R. Tarrío, and F. J. Ayala. Evolution of cis-regulatory regions versus codifying regions. *International Journal of Developmental Biology*, 47(7):665–673, 2003.
- [149] J. R. Roth and D. I. Andersson. Adaptive mutation: How growth under selection stimulates lac+ reversion by increasing target copy number. *Journal of Bacteriology*, 186(15):4855–4860, 2004.
- [150] J. R. Roth, E. Kugelberg, A. B. Reams, E. Kofoid, and D. I. Andersson. Origin of mutations under selection: the adaptive mutation controversy. *Annual Review of Microbiol*, 60:477–501, 2006.
- [151] F. Rothlauf and D. E. Goldberg. Redundant representations in evolutionary computation. *Evolutionary Computation*, 11(4):381–415, 2003.
- [152] D. Sankoff. Rearrangements and chromosomal evolution. *Current Opinion in Genetic and Development*, 13(6):583–587, 2003.
- [153] K. Sastry, U.-M. O’Reilly, and D. E. Goldberg. Population sizing for Genetic Programming based upon decision making. In *Genetic Programming Theory and Practice II*, pages 49–66. Kluwer Academic Publishers, 2004.
- [154] K. Schmitt. Using gene duplication and gene deletion in evolution strategies. Technical report, Collaborative Research Center 531, University of Dortmund, 2004.

- polymorphisms within gene regions: Phenotypic and evolutionary implications. *American Journal of Human Genetics*, 67(2):345–356, 2000.
- [187] Z. Yang and J. P. Bielawski. Statistical methods for detecting molecular adaptation. *Trends in Ecology and Evolution*, 15(12):496–503, 2000.
- [188] T. Yu. Program evolvability under environmental variations and neutrality. In *Proceedings of the 9th European Conference on Advances in Artificial Life (ECAL07)*, pages 835–844. Springer, 2007.
- [189] T. Yu and J. Miller. Neutrality and the evolvability of boolean function landscape. In *Proceedings of the Forth European Conference on Genetic Programming (EuroGP 2001)*, volume 2038 of *Lecture Notes in Computer Science*, pages 204–217. Springer, 2001.
- [190] Y. Yu, S. Murphy, and L. Murphy. A clustering approach to planning base station and relay station locations in IEEE 802.16j multi-hop relay networks. In *Proceedings of IEEE International Conference on Communications*, pages 2586–2591, Beijing, China, June 2008.
- [191] J. Zhang. Evolution by gene duplication: An update. *Trends in Ecology and Evolution*, 18(6):292–298, 2003.
- [192] J. Zhang, H. F. Rosenberg, and M. Nei. Positive Darwinian selection after gene duplication in primate ribonuclease genes. *Proceedings of the National Academy of Sciences*, 95(7):3708–3713, 1998.

