Design and Analysis of Multi-Parent Genetic Algorithms

Chuan-Kang Ting

Dissertation

in Computer Science

submitted to the

Faculty of Electrical Engineering,
Computer Science and Mathematics
University of Paderborn

in partial fulfillment of the requirements for the degree of

doctor rerum naturalium
(Dr. rer. nat.)

Paderborn, August 2005

To my parents, my wife Yu-Chin, and my daughter Julia.

獻給我的父母,妻子,和孩子

Acknowledgements

I am profoundly grateful to my advisor Professor Hans Kleine Büning for his constant encouragement, invaluable advice, thought-provoking questions, and unwavering support throughout my dissertation work. I would like to thank Professor Burkhard Monien and Professor Wilhelm Schäfer for their helpful suggestions and comments. I would also like to thank Professor Chungnan Lee and Professor Sheng-Tun Li, whose guidance has been the foundation of my advanced research in Germany.

I am deeply indebted to my colleague and friend Klaus Brinker for his thorough proofreading and helpful comments on this manuscript. On a personal level, I also thank him and his wife Kerstin for their friendship, hospitality, and encouragement in the past three years. Many thanks go to my colleagues in the Knowledge-based Systems research group for their kind help. I would also like to thank Oliver Kramer for the collaboration and discussions about evolutionary computation. Special thanks go to Yufeng Guo and Qiao Chen for their friendship and support.

I would like to thank the International Graduate School of Dynamic Intelligent Systems at the University of Paderborn for their excellent program and financial support. I am also indebted to the Graduate School team, especially to Eckhard Steffen and Astrid Canisius.

I owe my deepest gratitude to my family, particularly to my parents, for their unconditional love and unstinting support throughout my life. Moreover, I am very grateful for their help in taking care of my daughter. My daughter Julia, born during this research, has been a constant source of inspiration and deserves special credit. To my beloved partner in work and play, pleasures and pains, Yu-Chin, thank you for all the love, support, and happiness that you brought to me.

Vielen Dank an alle!

Chuan-Kang Ting
Paderborn, August 2005

Abstract

Multi-parent genetic algorithms (MPGAs) generalize genetic algorithms (GAs) by allowing more than two parents participating in crossover and have received a considerable number of satisfactory results. However, various issues arising from the increase of parents are still left open: What is the beneficial number of parents? Who should be mated? When will MPGAs outperform GAs? Why do MPGAs perform better?

This thesis addresses these open issues through design and analysis of MPGAs. First, a novel mating strategy is proposed to deal with the mating issue and the number of parents in MPGAs. Parents are filtered according to the tactics of tabu search for a balance of maintaining population diversity and supplying selection pressure. The resultant validity of mating is further used to adjust the number of parents adaptively. Consequently, the mating and the number of parents in MPGAs are well controlled. The experimental results on a series of common test functions show the superiority of this approach over GA and MPGA in terms of convergence speed and solution quality.

Second, we conduct a theoretical analysis on the influence of increasing the number of parents over the performance of crossover. The analysis focuses on *uniform scanning crossover* (U-Scan) and *occurrence based scanning crossover* (OB-Scan) since both of them are multi-parent generalizations of *uniform crossover* — the most popular crossover in GAs. A simple yet effective analytical model, the *uniform population* model, is presented as a systematic population environment for analysis. Under the assumption of uniform population, the analytical results reveal that on the one hand the number of parents in U-Scan exerts no influence on the performance of MPGAs; on the other hand, increasing the number of parents in OB-Scan will intensify the exploitation in MPGAs. Restated, raising parents in OB-Scan accelerates the convergence but is vulnerable to premature convergence. These theoretical claims are validated in an empirical manner.

Third, a Markov model for MPGAs is developed to understand the behavior of MPGAs, to investigate the causes of the superiority of MPGAs over GAs, and to discover the parameters that benefit MPGAs. Inceptively, we look into the variation of gene frequency affected by each MPGA operator. The integral influence of these operators is further formulated by Markov chain theory. Thus the proposed Markov model considers the separate as well as the integral effects of the population size, selection intensity, the number of parents, mutation rate, and generation gap in the course of evolution. Through this model, we show that U-Scan is essentially a special case of OB-Scan; precisely, U-Scan with any number of parents performs analogously with 2-parent OB-Scan, i.e. uniform crossover. Furthermore, we prove that the increase of parents in MPGAs using OB-Scan speeds up genetic drift and reinforces the bias of drift toward allele 1 or 0.

In addition, we make use of this Markov model to explore the convergence of MPGAs in the Generalized OneMax problems, including the OneMax problem and the BinInt problem. The analytical results of the mean fitness in these problems demonstrated that, as mutation is applied, MPGAs using OB-Scan gain preferable solution quality at the cost of convergence time. However, the defects do not outweigh the merits. Concerning the convergence speed, MPGAs using OB-Scan for particular numbers of parents are able to converge faster than GAs. Even in a short running time, MPGAs using n-parent OB-Scan with $n > 2$ are proved to be capable of better solution quality than GAs. The proposed Markov model, therefore, can explicitly predict the behavior of MPGAs and helps to find out the optimal setting of MPGAs for the Generalized OneMax problems. Furthermore, these outcomes can afford to hint the appropriate setting of MPGAs for other problem domains, which is of great benefit to extend and improve the utility of MPGAs.

Contents

List of Figures

List of Tables

Notation

$\mathbb{N}$	the set of natural numbers
$\mathbb{N}_{>1} = \{2, 3, 4, \cdots\}$	
$\mathbb{R}$	the set of real numbers
$\mathbb{R}_{+}$	the set of positive real numbers
$\mathbb{Z}$	the set of integers
$\mathbb{Z}_{*} = \{0, 1, 2, \cdots\}$	
$\bar{A}$	complement of set A
$\lvert A \rvert$	cardinality of set A
C	population
$\mathbf{c}$	chromosome
c_k	gene at locus k
l	chromosome length
m	population size
n	number of parents
γ_x	crossover rate
γ_m	mutation rate
γ_g	generation gap
η^{+}, η^{-}	control parameters of linear ranking selection
$U_{(k)}$	the k-order uniform population
$\mathcal{X}_{\mathrm{u}}$	uniform scanning crossover (U-Scan)
$\mathcal{X}_{\mathrm{ob}}$	occurrence based scanning crossover (OB-Scan)
$D(\mathbf{c})$	the set of distinct genes in chromosome $\mathbf{c}$
α	significance level
$\Pr$	probability measure
$\mathrm{E}[X]$	expectation of random variable X
$\mathrm{Var}(X)$	variance of random variable X
p_k	gene frequency of allele 1 at locus k
σ_k	variance of allele 1 at locus k
I	incomplete beta function
$\{G_k(t)\}$	Markov chain for gene frequency at locus k

$\{G(t)\}$	Markov chain for multiple gene frequencies
ρ_{ij}	transition probability from state i to state j
$\mathbf{P} = (\rho_{ij})$	transition matrix (also matrix of transition probability ρ_{ij})
$\boldsymbol{\pi}(t)$	state distribution at time t
$\boldsymbol{\pi}(0)$	initial distribution
$\boldsymbol{\pi}$	stationary distribution
$\mathbf{A}^k$	k^{th} power of matrix $\mathbf{A}$
$\mathbf{A}^{-1}$	inverse of matrix $\mathbf{A}$
$\mathrm{diag}(\mathbf{A})$	diagonal matrix with the same diagonal elements as $\mathbf{A}$
$\mathbf{I}$	identity matrix
$\mathbf{0}$	zero matrix
$\mathbf{F}$	fundamental matrix of an absorbing Markov chain
$\mathbf{Z}$	fundamental matrix of an ergodic Markov chain
μ_{ij}	mean time from state i to state j for an ergodic Markov chain
$\mathbf{M} = (\mu_{ij})$	matrix of mean time μ_{ij}
f_{ij}	first passage probability from state i to state j
δ_{ij}	Kronecker's delta
$\mathcal{I}$	selection intensity
$\mathcal{I}^{\mathrm{p}}$	selection intensity for gene frequency
β	initialization bias
$\bar{f}$	mean fitness of the population
$\bar{f}^s$	mean fitness of the selected parents
$\bar{f}^*$	mean convergence fitness of the population
σ_F^2	variance of fitness of the population
σ_F^{2*}	variance of convergence fitness of the population

1

Introduction

This thesis is about design and analysis of multi-parent genetic algorithms. Genetic algorithms (GAs) have received a great deal of attention on their effectiveness in a variety of optimization problems [43, 44, 48]. The basic idea of GAs is to enhance the candidate solutions of a given problem by simulating the mechanisms of natural evolution, such as selection, crossover, and mutation [55]. Crossover is the most salient operator in GAs. Traditionally, GAs adopt two parents in crossover to reproduce offspring. This idea is reasonable because, to the best of our knowledge, the form of sexual reproduction on Earth is absolutely of two parents. In computer world, *multi-parent crossovers* break through this natural limitation by allowing more than two parents in reproduction of offspring. *Multi-parent genetic algorithms* (MPGAs) represent the GAs using multi-parent crossover. With respect to the number of parents, MPGAs are generalizations of GAs.

Several multi-parent crossovers and MPGAs have been proposed and shown their superiority over GAs in a considerable number of problems [24, 25]. However, various issues arising from the increase of parents are still left open: What is the suitable number of parents for a given problem? Who should be mated? When will MPGAs outperform GAs? Why do MPGAs perform better? In this thesis, we will deal with these issues by means of design and analysis of MPGAs.

This introductory chapter recapitulates the algorithms GAs and MPGAs. Moreover, we will demonstrate the open issues in MPGAs and our goals. The contributions of this thesis will be shown afterwards. Finally, we will outline the organization of this thesis.

1.1 Genetic Algorithms

Genetic algorithms (GAs) are well-known meta-heuristic algorithms and have shown their effectiveness in a variety of fields, for instance, machine learning [48, 69], numerical optimization problems [20, 66], combinatorial problems [51, 82], multi-objective optimization problems [15, 21, 118], and design [7, 43]. The general principle of GAs is to simulate the mechanisms of natural evolution, such as selection, crossover, and mutation [55]. Based on Darwin's theory "Survival of the Fittest" [17], GAs are believed to be capable of evolving candidate solutions into better ones. To this end, GAs encode candidate solutions as *chromosomes*. Bit strings are widely used as the structure of chromosomes, while other structures, like real number, order (sequence), etc. are also applicable in GAs. Instead of a single chromosome, GAs evolve with a set of chromosomes, called the *population*. The fitness function is devised to evaluate the quality (*fitness*) of candidate solutions (chromosomes). Intuitively, for a maximization problem, the better the solution, the higher the fitness.

For binary-coded GAs, the chromosomes and the population are defined as follows.

Definition 1.1 (Chromosome and Population).

1. *A* chromosome $\mathbf{c}$ *is encoded as a bit string, i.e.* $\mathbf{c} \overset{\text{def}}{=} (c_1, \dots, c_l) \in \{0,1\}^l$*, where* c_i *denotes a* gene *and* l *is the* chromosome length.
2. *The* population C *is a set of chromosomes:* $C \overset{\text{def}}{=} \{\mathbf{c}_1, \dots, \mathbf{c}_m\}$*, where* $\mathbf{c}_i \in \{0,1\}^l$ *and* m *is the* population size.

Here we clarify the notion of 'locus–gene–allele' used in this thesis.

Definition 1.2 (Locus–Gene–Allele). *For a* gene $c_i \in \{0,1\}$*, the index (position)* i *denotes the* locus *of* c_i*, and the potential values* 0 *and* 1 *are both* alleles *of* c_i.

Figure 1.1 demonstrates the process of GAs. The evolution of GAs begins with the initialization of the population. Afterwards, GAs embark on the process of reproduction. First, the *selection* operator picks two chromosomes from the population to serve as *parents*. Next, GAs perform *crossover* on these two parents to reproduce their *offspring*. The predetermined probability, crossover rate, defines the probability to perform crossover. Analogously, *mutation* is performed with a probability, mutation rate, on the offspring reproduced by crossover to slightly alter some genes. This process of reproduction repeats until the set of offspring, called the *subpopulation*, is filled. Acting on "Survival of the Fittest", the *survivor* operator draws the fittest chromosomes out of the subpopulation with (or without) the primitive population; the chosen chromosomes will constitute the population for the next generation. Following this procedure, GAs evolve until a predetermined termination criterion is met. The pseudocode of GAs is given in Algorithm 1.

Crossover (or *recombination*) is the most salient feature of GAs. It reproduces offspring by exchanging and recombining genetic material from two parents. Figure 1.2 illustrates three common crossovers: *one-point crossover* [20, 55], *two-point crossover* [20, 55], and *uniform crossover* [99]. One-point crossover divides each parent into two sections by one crossover point; then it exchanges and recombines one of the two sections with another parent. Similarly, two-parent crossover divides each parent into three segments by two crossover points and exchanges the middle section with another parent. Instead of sectionally, uniform crossover treats each locus independently and determines the donor at random. It is deserving of note that one-point, two-point,

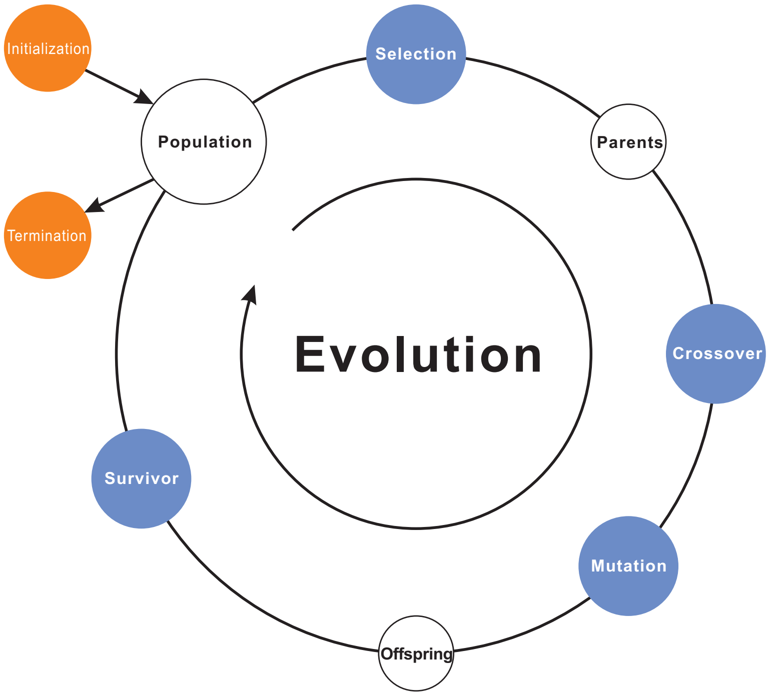


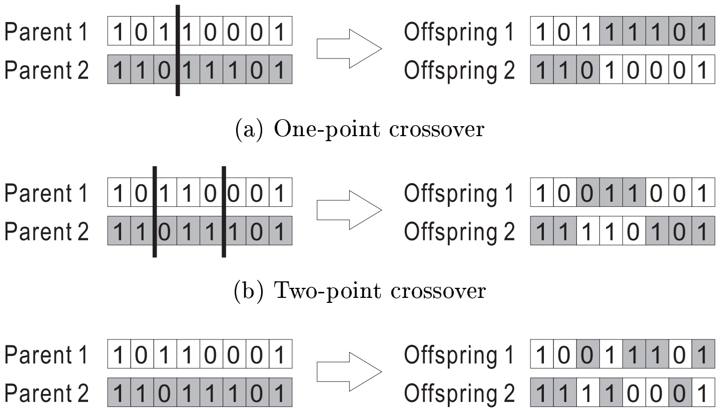
Fig. 1.1. Process of GAs

Algorithm 1 Genetic Algorithms

```
$t \leftarrow 0$
$Initialize$ population $C(t)$
$Evaluate$ $C(t)$
repeat
  repeat
    $\mathbf{c}_1, \mathbf{c}_2 \leftarrow Select(C(t))$
    $\mathbf{c}'_1, \mathbf{c}'_2 \leftarrow Crossover(\mathbf{c}_1, \mathbf{c}_2)$
    $\mathbf{c}''_1, \mathbf{c}''_2 \leftarrow Mutate(\mathbf{c}'_1, \mathbf{c}'_2)$
    $C(t+1) \leftarrow C(t+1) \cup \{\mathbf{c}''_1, \mathbf{c}''_2\}$
  until ($C(t+1)$ is filled)
  $C(t+1) \leftarrow Survivor(C(t), C(t+1))$
  $t \leftarrow t+1$
until (termination criterion is satisfied)
```

and even multi-point crossovers are special cases of uniform crossover. In other words, uniform crossover is a generalization of these k-point crossovers.

The way of exchanging material in crossover implies that crossover is subject to the representation of chromosomes. The three crossovers shown in Fig. 1.2, for example, are applicable to binary-coded GAs as well as real-coded GAs, but they are inapplicable to order-based GAs. For order-based GAs, some other crossovers are proposed, such as *partially mapped crossover* (PMX) [49], *order crossover* (OX) [18], *cycle crossover* (CX) [78], and the series of *edge recombination* (EX) [23, 65, 76, 98, 102, 115]. On the


Parent 1 1 0 1 1 0 0 0 1
Parent 2 1 1 0 1 1 1 0 1
Offspring 1 1 0 1 1 1 1 0 1
Offspring 2 1 1 0 1 0 0 0 1
Parent 1 1 0 1 1 0 0 0 1
Parent 2 1 1 0 1 1 1 0 1
Offspring 1 1 0 0 1 1 0 0 1
Offspring 2 1 1 1 1 0 1 0 1
Parent 1 1 0 1 1 0 0 0 1
Parent 2 1 1 0 1 1 1 0 1
Offspring 1 1 0 0 1 1 1 0 1
Offspring 2 1 1 1 1 0 0 0 1

(a) One-point crossover

(b) Two-point crossover

(c) Uniform crossover

Fig. 1.2. One-point crossover, two-point crossover, and uniform crossover

other hand, these crossovers are not suitable for binary-coded GAs. Regarding the key role of crossover in the performance of GAs, a growing number of crossovers [7, 95] are devised to improve the performance or to handle specific problems and representation.

The performance of GAs is mostly examined in an empirical manner owing to their intrinsically stochastic property. The development of theory, nevertheless, is a paramount research topic in GAs. Several theoretical methods have been proposed to substantiate the merits of GAs or introduced to describe and predict the behavior of GAs. In the following paragraphs we briefly describe some important methods of them. For an overview of theoretical aspects of GAs refer to [31, 58, 86, 88, 91, 92].

1. **Schema Theorem:**
   This theorem was first proposed by Holland [55] and was deemed as the fundamental to explain the power of GAs. A *schema* stands for a hyperplane in the search space; the order of a schema is the number of definite positions (0 or 1) in the hyperplane. The schema theorem shows that a GA processes nearly $O(m^3)$ schemata at each generation; this phenomenon leads to the notion of *implicit parallelism* in GAs. The *building block hypothesis* (BBH) [48] further states that GAs can yield long and high-order schemata by progressively combining short low-order ones. However, the schema theorem is controversial [4, 70, 80] and cannot describe the dynamic behavior of GAs.
2. **Markov Chain Theory:**
   The process of GAs is only relevant to the previous generation but is irrelevant to the events earlier than the previous generation. In consideration of this memoryless property, Markov chain theory is particularly appropriate for modeling the behavior of GAs [5, 13, 50, 56, 64, 100] and for analyzing the global convergence of GAs [2, 19, 26, 77, 85, 113]. Even though Markov chain theory serves as a powerful

tool to describe and predict the behavior of GAs in theory, the enormous degree of transition matrix hinders its applicability in practice: the transition matrix in Nix and Vose's model [77, 113], for example, has $\binom{m+2^l-1}{2^l-1}$ columns and rows.

3. **Quantitative Genetics:**
The concept of GAs is inspired by natural evolution; thus, it is intuitive to analyze GAs by means of existing analytical methods in genetics. Through theories in quantitative genetics, Mühlenbein [72, 73] developed a theory to analyze the time complexity and to predict the behavior of a specific GA — the *breeder genetic algorithm* (BGA). The defects of this theory include the assumption of spherical symmetry for the fitness and the assumption of infinite population, both of which are inapplicable to conventional GAs [31, 87, 88].

4. **No Free Lunch Theorem (NFL):**
The no free lunch theorem [117] states that all optimization methods have equivalent performance, if averaged over all fitness functions. In other worlds, no GA can outperform any other optimization methods in *all* kinds of problems. This theorem reveals that any evolutionary algorithm has its *cost* and *limit* in its superiority of performance. Nonetheless, intense and contentious debate over the utility of NFL theorem was sparked and is still ongoing in the evolutionary computation community: First, the average performance over all fitness functions is of little interest in practice [22, 88]. Second, the NFL assumes no revisiting of the same point in the search space, which does not hold in GAs unless some memory mechanism is adopted.

1.2 Multi-Parent Genetic Algorithms

Traditionally, GAs adopt two parents in crossover to reproduce offspring. This idea is reasonable because, to the best of our knowledge, the form of sexual reproduction on Earth is absolutely of two parents. However, in computer world it is feasible to break through this natural limitation. Multi-parent crossovers are the very crossovers that break through the '2-parent' confines and allow for operation with more than two parents.

Multi-parent genetic algorithms (MPGAs) are genetic algorithms using multi-parent crossover. Simply speaking,

$$\textbf{MPGAs} = \textbf{GAs} + \textbf{Multi-Parent Crossover}.$$

With respect to the number of parents, MPGAs are generalizations of GAs.

The debut of multi-parent crossovers in evolutionary computation was as *global recombination* in evolutionary strategies (ES) [9, 93]. Global recombination determines each offspring gene according to the whole population rather than only two parents. Such a global form, as a result, allows an offspring to inherit from more than two parents. For GAs, several multi-parent crossovers were proposed afterwards. By generalizing uniform crossover and one-point crossover, Eiben et al. proposed two multi-parent crossovers: *scanning crossover* [30] and *diagonal crossover* [35, 36]. Depending on the heuristics applied to it, scanning crossover has three variations: *uniform scanning crossover* (U-Scan), *occurrence based scanning crossover* (OB-Scan), and *fitness based scanning crossover* (FB-Scan). Experimental results [30, 34, 35, 36] on several test functions show that, in terms of success rate, both scanning crossover and diagonal crossover outperform their 2-parent versions, namely uniform crossover and one-point crossover. The results also indicate that there exists a positive correlation between the success rate and the number of parents in diagonal crossover. Such a correlation, nevertheless, does not hold in scanning crossover. In terms of convergence speed, the superiority of these two multi-parent crossovers over their 2-parents versions is inconclusive. Eiben and Bäck [27, 28] further examined the performance of these two multi-parent crossovers in ES. They concluded that multi-parent crossovers are practically sound for ES from experimental results on seven numerical problems.

Aside from diagonal crossover and scanning crossover, Mühlenbein et al. [74, 112] introduced the concept of global recombination into GAs as *gene pool recombination* (GPR). Rather than from two parents, GPR samples the genes for crossover from the gene pool, which consists of several pre-selected parents. The studies show that GPR and its variants are easier to analyze and these methods can converge faster than 2-parent recombination. Tsutsui and Jain [110] proposed *multi-cut crossover* (MX) and *seed crossover* (SX). Noteworthily, multi-cut crossover generalizes the classic two-point crossover and was shown empirically to outperform diagonal crossover.

Some multi-parent crossovers for the GAs using representation other than bit string have been proposed as well. For real-coded GAs, Tsutsui and Ghosh [108, 109] presented a series of multi-parent crossovers: *center of mass crossover* (CMX), *multi-parent feature-wise crossover* (MFX), and *seed crossover* (SX). Experimental results demonstrate that these multi-parent crossovers can lead to better performance although the performance is problem-dependent. Another multi-parent crossover, *simplex crossover* (SPX) [111], reproduces by the simplex sampled from multiple parents. Their results show that this method performs well with three or four parents for multimodal and epistatic problems. Moreover, Kita et al. [61] introduced multiple parents into *unimodal normal distribution crossover* (UNDX) to enhance the diversity of offspring. This multi-parent extension of UNDX exhibits its improvement in search ability on highly epistatic problems.

For order-based GAs, Eiben et al. [30] devised *adjacency based crossover* (ABC), which is similar to *edge recombination* [115, 116] but extends the number of parents to an arbitrary value. In addition, some of the multi-parent crossovers for binary-coded GAs can be directly applied to order-based GAs. For example, scanning crossover [30] was adopted to solve the graph coloring problem and the traveling salesman problem (TSP). However, experimental results point out that the benefit of using more than two parents in either ABC or scanning crossover is inconclusive.

To tackle multi-objective optimization problems, Lis and Eiben [63] introduced the concept of sex into multi-parent crossover as *multisexual genetic algorithm* (MSGA). The sex is appended to each chromosome in order to indicate a specific criterion to optimize. Restated, the fitness is evaluated by the objective function corresponding to the sex. Esquivel et al. [38, 39] extended this method into the *multi-sexual-parents-crossover genetic algorithm* (MSPC-GA) by enabling multiple parents per sex and multiple crossovers per mating. This approach obtains a satisfactory result in the number of non-dominated solutions on the Pareto front. Leiva et al. [62] further incorporated this approach with local search and received even better results.

A few studies aim for theoretical analysis on multi-parent crossovers. Concerning diagonal crossover and scanning crossover, Eiben et al. [36] investigated the effects of these two crossovers on distributions by Walsh product. For a GA using 10-bit representation and no mutation, their analytical results indicate that the performance of 3-parent uniform scanning crossover is very close to that of 3-parent diagonal crossover; specifically, their difference in the expected fitness on the inverted Rastrigin function is less than 1. In addition, Eiben and Schippers [32] looked into the impact of the number of parents on NK-landscape [59] and confirmed the advantage of using more parents in mildly epistatic problems: The performance of diagonal crossover improves with the increase of parents, and yet this correlation does not hold in uniform scanning crossover. The number of *donors*, namely the parents really participating in the process of crossover, is further examined in [96]. This analysis shows that almost all parents will be donors if the number of parents is relatively small to the chromosome length; however, for short chromosome length only a fraction of parents will become donors. Moreover, Schippers [90] carried out a study on the genetic drift of scanning crossovers. Genetic drift [53] is a phenomenon that in a finite population without external forces (e.g. selection and mutation) the genetic variability of a locus will decay with time and eventually get fixed to some allele. Schippers proved that uniform scanning crossover has no influence on genetic drift whilst occurrence based scanning crossover induces severe genetic drift as the number of parents is increased.

The above-mentioned research on practical and theoretical aspects of MPGAs showed the power of MPGAs in optimization problems and demonstrated some characteristics of MPGAs. Nonetheless, various issues arising from the increase of parents in crossover remain open: What is the suitable number of parents? Who should be mated? When will MPGAs outperform GAs? Why do MPGAs perform better? Section 1.3 will shed light on these issues and present our goals for them. In addition, this thesis is concerned with *diagonal crossover* and *scanning crossover* since these two crossovers generalize the most common 2-parent crossovers: one-point crossover and uniform crossover, respectively. The analysis of MPGAs, in particular, concentrates on *scanning crossover* in that this crossover further generalizes diagonal crossover. More detailed descriptions and formal definitions of diagonal crossover and scanning crossovers, including U-Scan and OB-Scan, are given below.

1.2.1 Diagonal Crossover

Diagonal crossover was first proposed by Eiben [35, 36] as a generalization of one-point crossover. For n parents, diagonal crossover divides each parent into n sections by (n–1) crossover points. Thereafter the crossover picks one section respectively from each parent in a diagonal way and recombines these sections into a complete offspring. This manner of picking sections gives the name 'diagonal' crossover. Figure 1.3 illustrates 3-parent diagonal crossover and 2-parent diagonal crossover, i.e. one-point crossover. Note that in these examples, one execution of diagonal crossover reproduces only one child. Clearly, diagonal crossover with 2 parents coincides with one-point crossover; that is, diagonal crossover generalizes one-point crossover.

Definition 1.3 (Diagonal Crossover). *Given n parents $\mathbf{c}_1, \ldots, \mathbf{c}_n \in C$ and the crossover points $x_1, \ldots, x_{n-1} \in \{1, \ldots, l\}$ with $x_1 < x_2 < \cdots < x_{n-1}$. Diagonal crossover reproduces the offspring $\mathbf{c}' = (c'_1, \ldots, c'_l)$ by*

$$c'_k = (\mathbf{c}_j)_k \quad \textit{for } x_{j-1} < k \leq x_j,$$

where $(\mathbf{c}_j)_k$ denotes the k^{th} gene of parent $\mathbf{c}_j$, $x_0 = 1$ and $x_n = l$.

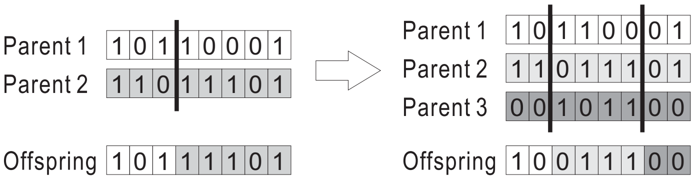


Fig. 1.3. Examples of 2-parent diagonal crossover (left) and 3-parent diagonal crossover (right)

1.2.2 Uniform Scanning Crossover (U-Scan)

Uniform scanning crossover is a multi-parent generalization of uniform crossover. In uniform crossover, the donor to an offspring gene is randomly chosen from two parents. Analogously, U-Scan determines the donor at random but extends the number of parents from two to an arbitrary number larger than one. Owing to the random manner of choosing donors, each parent has equal likelihood to give its gene in both uniform crossover and U-Scan. Figure 1.4 illustrates 2-parent U-Scan and 4-parent U-Scan, where 2-parent U-Scan corresponds to uniform crossover. The formal definition of U-Scan is given as follows.

Definition 1.4 (U-Scan). *Given n parents $\mathbf{c}_1, \ldots, \mathbf{c}_n \in C$. U-Scan reproduces the offspring $\mathbf{c}' = (c'_1, \ldots, c'_l)$ by*

$$c'_k = (\mathbf{c}_j)_k \quad \textit{for } k = 1, \ldots, l,$$

where $(\mathbf{c}_j)_k$ denotes the k^{th} gene of parent $\mathbf{c}_j$ and $j \in \{1, \ldots, n\}$ is generated randomly for each k.

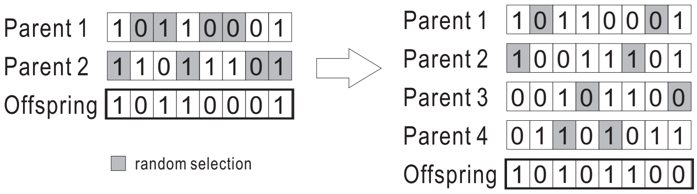


Fig. 1.4. Examples of 2-parent U-Scan (left) and 4-parent U-Scan (right)

1.2.3 Occurrence Based Scanning Crossover (OB-Scan)

Occurrence based scanning crossover is another multi-parent generalization of uniform crossover. Rather than random, OB-Scan determines offspring genes depending on the occurrence of parental genes at that locus. Specifically, it picks the majority of parental genes as the offspring gene for each locus. Note that in this thesis OB-Scan is defined to break ties by randomly[1] choosing a binary number. Figure 1.5 gives examples of 2-parent OB-Scan (corresponding to uniform crossover) and 4-parent OB-Scan.

Definition 1.5 (OB-Scan). *Given n parents $\mathbf{c}_1, \ldots, \mathbf{c}_n \in C$. OB-Scan reproduces the offspring $\mathbf{c}' = (c'_1, \ldots, c'_l)$ by*

$$c'_k = \begin{cases} 0 & \textit{if } \sum_{j=1}^{n} (\mathbf{c}_j)_k < \frac{n}{2} \\ 1 & \textit{if } \sum_{j=1}^{n} (\mathbf{c}_j)_k > \frac{n}{2} \\ \mathrm{Rand}(0,1) & \textit{otherwise} \end{cases} \qquad \textit{for } k = 1, \ldots, l,$$

where $(\mathbf{c}_j)_k$ denotes the k^{th} gene of parent $\mathbf{c}_j$ and $\mathrm{Rand}\,(0,1) \in \{0,1\}$ is a binary random function.

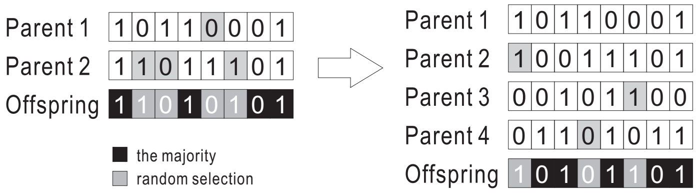


Fig. 1.5. Examples of 2-parent OB-Scan (left) and 4-parent OB-Scan (right)

[1] The original OB-Scan [30] breaks ties by directly inheriting the gene from the first selected parent. However, random tie break conforms to the generalization of uniform crossover.

1.3 Issues and Goals

The fundamental issue of MPGAs is:

Can GAs benefit from the increase of parents?

Most of the existing work on MPGAs, as reviewed in the previous section, focuses on design of multi-parent crossover for more than two parents to participate in crossover, and ordinarily verifies the benefit of the resultant MPGA in an empirical manner. However, various issues arising from the increase of parents in GAs are essential but still left open:

- What is the suitable number of parents?
- Who should be mated?
- When will MPGAs outperform GAs?
- Why do MPGAs perform better?

The present thesis aims to resolve these open issues. According to their genre, these issues are categorized as issues in *design* of MPGAs and issues in *analysis* of MPGAs. The following sections will discuss these issues and present our goals for them.

1.3.1 Issues and Goals in Design of MPGAs

Issue 1: *What is the suitable number of parents?*

This issue is a consequential issue of raising the number of parents. Many studies, as reviewed in Section 1.2, have indicated that the number of parents plays an important role in the performance of MPGAs, and a lot of experiments have been conducted to find the suitable number of parents for particular test functions. However, like other parameters in GAs, e.g. population size and mutation rate, there is no winner of the number of parents for all problems. That is to say, the suitable number of parents is problem-dependent. In addition, this number is subject to the setting of other MPGA operators since the performance of MPGAs is an integral effect of all operators. Thus, this issue also manifests the importance of the third and the fourth issues, which help to find the suitable number of parents for a given problem with regard to a setting of MPGA operators.

The first goal of this thesis is to establish a rule to determine the number of parents that is beneficial to the performance of MPGAs. Considering the fruitfulness of adaptive approaches on tuning the GA parameters [29, 54, 97], an adaptive method to adjust the number of parents during evolution is preferred. To this end, we propose to adjust the number of parents in response to the status of population diversity. Through the adaptiveness, this method can overcome the difficulty in determining the appropriate number of parents for different problems.

Issue 2: *Who should be mated?*

In the conventional GAs, no mating strategy is applied to the results of selection; in other words, parents are approved without any further examination after they are selected at random or by fitness. Indeed, mating in nature is more complicated. The mating of human beings, for example, is restricted by the blood relationship, which is irrelevant to fitness. Whitley [114] pointed out that a well-balanced mating strategy regarding population diversity and selection pressure can substantially improve GAs. In MPGAs, the increase of parents complicates the situation of mating, and the role of mating becomes more crucial. Hence a well-designed mating strategy will be advantageous to the performance of MPGAs.

The second goal is to develop a mating strategy for MPGAs in order to improve their performance. We will implant the tactics of tabu search in the mating of MPGAs. As a result, parents are filtered in thoughtful consideration of the balance between maintaining population diversity and supplying selection pressure. Additionally, this mating strategy will utilize the rule designed for the first goal in order to adaptively adjust the number of parents in MPGAs. The mating strategy will be amenable to indicating the ideal mates and determining the appropriate number of parents for different problems. Consequently, an improvement in performance of MPGAs will be achieved.

1.3.2 Issues and Goals in Analysis of MPGAs

Issue 3: *When will MPGAs outperform GAs?*

This issue concerns the interaction between multi-parent crossover and other MPGA operators. In addition to crossover, MPGA (or GA) operators include selection, mutation, and survivor. The performance of MPGAs, therefore, is attributed to not only the crossover but also the synergy of all operators. Let us consider the exploitation and the exploration of evolutionary search. Generally, selection affects the evolutionary search in both exploitation and exploration, crossover can be exploitative or exploratory, and mutation focuses on exploration. That is, each operator has it own effect on the evolutionary search, and this effect also interacts with other operators. Thus, to investigate the influence of raising parents upon MPGAs, we need to take all the properties of genetic operators into account, in addition to multi-parent crossover.

The third goal is to model the behavior of MPGAs. A theoretical analysis will be conducted on the variation of gene frequency caused by selection, crossover, mutation, and survivor, respectively. Moreover, we will build a Markov model to formulate the variation of gene frequency over time, from which the mean fitness can be derived. Consequently, the individual and the integral effects of all MPGA operators are derived. Instead of empirically, this model provides theoretical evidence when MPGAs are able to outperform GAs. Through this model we can determine the suitable number of parents associated with the setting of other MPGA operators, such as selection bias and mutation rate.

Issue 4: *Why do MPGAs perform better?*

Many MPGAs have been shown to outperform GAs; nonetheless, the effectiveness of them is mostly validated in an empirical manner. The characteristics and the impact of raising parents on GAs are still left open in theory. Without knowledge about them, we can merely choose the number of parents through trial and error when applying MPGAs. This time-consuming job by no means meets our expectation of MPGAs. Theoretical analysis on multi-parent crossovers, therefore, is not only essential to enhance the knowledge about the behavior of MPGAs but also valuable for the application of MPGAs.

The fourth goal of this thesis is to theoretically analyze the influence of raising the number of parents on MPGAs. To this end, we will investigate the likelihood for a multi-parent crossover to reproduce better, equal, or worse offspring, and the impact of increasing the number of parents on it. As a result, the correlation between the number of parents and the performance of multi-parent crossover can be explored. The resultant likelihood also contributes to characterize the behavior of MPGAs and to explore the advantages or disadvantages of applying more parents in MPGAs. In addition, the proposed Markov model will be used to examine the genetic drift in MPGAs. This will provide an insight into how fast the population diversity in MPGAs is lost with respect to the adopted number of parents.

1.4 Contributions

In meeting the above goals, the present thesis makes some solid contributions to the theoretical as well as the practical aspects of MPGAs. Moreover, since GAs are special cases of MPGAs, the contributions of this thesis are also applicable to conventional GAs.

- **Theoretical Foundations:** In this thesis we propose the uniform population model for theoretical analysis of crossover, investigate the variation of gene frequency caused by each MPGA operator, and build a Markov model for MPGAs. First, the uniform population model establishes a simple yet effective way to analyze in theory the probability for a crossover to reproduce better, equal, or worse offspring. Second, beyond the constraint of uniformity in the population, the variation of gene frequency is investigated and the outcome successfully specifies the respective effects of MPGA operators on gene frequency. Finally, we formulate by Markov chain theory the integral influence of MPGA operators upon the gene frequency in the course of evolution. The proposed Markov model can give the expected fitness over time and predict the mean convergence fitness and the mean convergence time. All the theoretical results, furthermore, are examined empirically and show a high level of consistency with the experimental results.
- **Characterization of the Role of Parents in MPGAs:** Two characters of MPGAs are investigated: the *exploitation* and the *genetic drift*. The analysis based on

uniform population model clues the impact of multi-parent crossover on exploitation of MPGAs. Moreover, the analysis based on Markov chain theory gives the expected time (generations) that population diversity drains away and the probability to drift into allele 1 or allele 0.

- **Predictive Model for MPGAs:** This thesis builds a Markov model for MPGAs, which concerns the integral influence of the *population size*, the *selection intensity* in selection, the *number of parents* in crossover, the *mutation rate* in mutation, and the *generation gap* in survivor over generations. This Markov model can precisely predict the behavior of MPGAs and helps to find the optimal setting for the above parameters in MPGAs.
- **Improvement of Performance:** A mating strategy for MPGAs is proposed to deal with the mating issue and the number of parents. First, this mating strategy sifts the parents for a balance between population diversity and selection pressure. Second, it adjusts the number of parents in response to the diversity in the population. The MPGA using this mating strategy is shown to be able to improve the original MPGA in solution quality and convergence speed. Moreover, the presented theoretical analyses characterize the behavior of MPGAs and give guidelines for parameter setting — both help to enhance the performance of MPGAs.
- **Guidelines for Determining the Optimal Parameters in MPGAs:** We achieve this goal by means of design and analysis of MPGAs. Depending on design, the MPGA using the proposed mating strategy adaptively controls the number of parents according to the situation of the population. On the other hand, theoretical analysis gives useful hints on determining the appropriate number of parents. The above-mentioned Markov model is helpful for finding the optimal setting for population size, selection intensity, mutation rate, and generation gap in MPGAs for the Generalized OneMax problem. These outcomes also clue the appropriate parameter setting of MPGAs for other problem domains.

1.5 Organization

The remainder of this thesis is organized as follows. Chapter 2 proposes a mating strategy to cope with the mating issue and the number of parents in MPGAs. The proposed mating strategy sorts out the parents who are beneficial to balance population diversity and selection pressure. An accompanied rule is further established to adaptively reduce the number of parents according to the remaining diversity of the population. As a result, this mating strategy achieves a harmony in exploitation and exploration of MPGAs and adjusts the number of parents adaptively. A series of experiments will be conducted to inspect the merits of this strategy.

In Chapter 3 and Chapter 4, we perform theoretical analysis to explore the characteristics of MPGAs and to investigate the whys and the wherefores that MPGAs outperform GAs. Chapter 3 analyzes the performance of multi-parent crossover under the assumption of a specific population model — the uniform population model. A criterion based on uniform population is further presented to evaluate the performance of crossover. Accordingly, we conduct analysis on two multi-parent crossovers: uniform scanning crossover (U-Scan) and occurrence based scanning crossover (OB-Scan). The analysis calculates the probabilities for a crossover to reproduce better, equal, or worse offspring. These probabilities serve to identify how effectively a crossover leads the evolutionary search toward promising region. In addition, they imply the tendency of crossover toward exploitation or exploration. We will carry out experiments on common test functions to verify the capability of this analysis.

Chapter 4 builds a Markov model based on the variation of gene frequency in MPGAs. First, we look into the respective effects of MPGA operators on gene frequency. Next, we model by Markov chain theory the integral influence of these operators over generations. The proposed Markov model, therefore, concerns the separate as well as the integral effects of the *selection intensity* in selection, the *number of parents* in crossover, the *mutation rate* in mutation, and the *generation gap* in survivor in the course of evolution. Furthermore, we apply this Markov model to analyze two aspects of MPGAs: the genetic drift and the convergence. The theoretical results will be examined by a series of experiments.

Finally, Chapter 5 draws the conclusions including the summary of each chapter, the contributions of this thesis, and the directions for future work.

2

Design for MPGAs

This chapter proposes a mating strategy based on tabu search to deal with two issues in MPGAs:

- Whom are ideal to mate with?
- What is the suitable number of parents?

The proposed algorithm, called tabu multi-parent genetic algorithm (TMPGA), utilizes the tabu restriction and the aspiration criterion of tabu search to sift parents in consideration of population diversity and selection pressure. The resultant validity of mating is further used to adaptively adjust the number of parents participating in mating. Consequently, the disruptiveness caused by the increase of parents in crossover is controlled. Experimental results on four common test functions show that TMPGA can achieve better performance than both GA and MPGA in convergence speed and solution quality.

The present chapter is based on the work [105, 107].

2.1 Introduction

Multi-parent genetic algorithms have shown their effectiveness in a considerable number of optimization problems. However, there exist two issues of applying multi-parent crossovers: the number of parents and the selection of mates. Concerning the number of parents, MPGAs break through the limitation of using two parents but commonly set the number of parents fixed. Adaptive methods for 2-parent crossovers or mutations have received a lot of satisfying results [29, 54, 97]; thus, it is promising to tune the number of parents adaptively in MPGAs.

Conventionally, no mating strategy is applied to the results of selection in GAs; that is to say, parents are approved without any further examination after they are selected at random or just by fitness. Indeed, mating in nature is more complicated. For example, with respect to human beings, as well as specific factors such wealth and appearance that objectively contribute to fitness, some implicit factors also potentially guide mating, such as blood relationship. Inspired by this observation, a number of mating strategies [107] for GAs have been proposed and shown success stories in dealing with population diversity and selection pressure. Nevertheless, the increase of parents complicates the situation of mating. In addition, a notable augmentation of disruptiveness arises from the increase of parents participating in crossover. This disruptiveness on the one hand leads to a more diverse exploration, which helps to prevent premature convergence. On the other hand, it slows down convergence speed. Therefore, a well-balanced mating strategy in maintaining population diversity and supplying selection pressure is needed to enhance the performance of GAs as well as MPGAs.

In this chapter, we propose the *tabu multi-parent genetic algorithm* (TMPGA) to deal with the above issues by integrating tabu search (TS) into MPGAs. Instead of running the two algorithms alternatively, TMPGA implants the characteristics of TS in MPGA's mating strategy: First, the tabu list, the memory structure in TS, is used to record the trajectory of evolution and to preclude certain chromosomes from mating in consideration of diversification. Second, the aspiration criterion is applied to supply selection pressure for intensification under the restriction of tabu. The advantage of such a strategy has been validated for 2-parent crossover to harmonize selection pressure and population diversity [106, 107]. In TMPGA, the tabu strategy further adjusts the number of parents according to the condition of mating pool. As a result, the disruptiveness caused by multi-parent crossover is depressed. Several experiments will be conducted to examine the effectiveness of TMPGA in comparison to GA and MPGA.

The rest of this chapter is organized as follows. Section 2.2 gives a brief description of TS. In Section 2.3 we describe the proposed TMPGA in detail. Section 2.4 presents experimental results of TMPGA. Finally, conclusions are drawn in Section 2.5.

2.2 Tabu Search

Tabu search (TS) is a meta-heuristic algorithm and has shown its power in a lot of optimization problems, especially in combinatorial problems [45, 46, 47]. This approach uses explicit memory structures to record the search trajectory. According to its "forbiddance-versus-permission" tactic based on the recorded information, TS guides the search in consideration of both intensification and diversification. The basics of TS are listed below.

1. *Move*: The process from a solution to another one, e.g. the process from the current solution to its neighbor.
2. *Neighborhood*: The set of candidate solutions related to the current solution. The hamming distance is commonly used to define the range of neighborhood.
3. *Tabu list*: A memory structure used to record the forbidden moves. This is the most salient feature of TS. The concept of tabu list is to prevent the search from being mired in the local optima by indicating certain moves are *tabu*, i.e. forbidden. The size of tabu list affects the level of restriction to the search. A large size of tabu list encourages the search to explore unvisited territory, that is, diversification [3].

4. *Aspiration criterion*: The very criterion to override the tabu restriction. Specifically, it allows the superior solution to be chosen despite the restriction of tabu. This criterion supports the ability of intensification for the search.

As Algorithm 2 shows, TS begins with an initial candidate solution, which is ordinarily generated at random. Then the generate-and-test process follows. First, it generates the neighborhood $N(x)$ of the current solution x, and sorts them into a descent order for maximization problems. From the best neighbor to the worst one, TS checks the neighboring solutions one by one. A solution is acceptable only if it meets the two conditions: it is not in the tabu list or it satisfies the aspiration criterion. Once an acceptable neighbor x' is found, this check will be terminated and the neighbor will be passed to next step as the current solution. Meanwhile, the move from x to x' is included in the tabu list. This process is repeated until the predetermined maximal number of runs is reached; the best solution obtained so far is the output of this algorithm.

Algorithm 2 Tabu search (for maximization problems)

```
Choose $x \in X$
$T \leftarrow \phi$
$t \leftarrow 0$
$x^* \leftarrow x$
repeat
  Generate neighborhood $N(x)$
  Sort elements in $N(x)$: $N(x) = \{x'_1, \dots, x'_n\}$ with $f(x'_1) > \cdots > f(x'_n)$
  Find the smallest $k \in \{1, \dots, n\}$ such that $x'_k \notin T$ or $f(x'_k) > f(x^*)$
  $x \leftarrow x'_k$
  $T \leftarrow T \cup \{x'_k\}$
  $x^* \leftarrow \max(x^*, x)$
  $t \leftarrow t + 1$
until $(t > t_{\mathrm{Max}})$
```

2.3 Tabu Multi-Parent Genetic Algorithm (TMPGA)

The tabu multi-parent genetic algorithm (TMPGA) integrates the strategy of TS into the mating strategy of MPGA. First, the tabu list restricts the mating in an *incest-prevention* manner to maintain diversity. Second, the aspiration criterion releases the tabu restriction on mating in order to supply selection pressure. Such a synergy of the tabu list and the aspiration criterion is expected to achieve a harmony in maintaining population diversity and supplying selection pressure [107]. Moreover, the outcome of mating affects the number of parents participating in multi-parent crossover. The TMPGA therefore addresses the issues of the mating and the number of parents. However, some modifications to MPGA are necessary in order to incorporate the strategy of TS into it. The following sections will elaborate on the components and the algorithm of TMPGA.

2.3.1 Representation

To accommodate the memory structure of TS to GA, two components are appended to the representation of chromosomes. First, a *clan* number is introduced to identify chromosomes. This number is assigned uniquely during initialization. In the process of reproduction, offspring will inherit the clan number from one of their parents randomly. Second, the tabu list, which records a set of forbidden clans to mate with, is appended to the structure of chromosomes as well. A formal definition is given as follows.

Definition 2.1 (Representation in TMPGA). *Let $G = (c_1, \dots, c_l)$ be the genes of chromosomes in GAs, σ be the clan number, and $T = (\tau_1, \dots, \tau_\nu)$ be the tabu list of size ν. In TMPGA, the chromosomes are represented as*

$$\mathbf{c} \overset{\text{def}}{=} (G, \sigma, T) = (c_1, \dots, c_l, \sigma, \tau_1, \dots, \tau_\nu).$$

Example 2.2. Figure 2.1 illustrates a representation for binary-coded TMPGA. The genes are encoded in bit string $G = (1, 0, 1, \dots, 0)$ concerning the solutions of the given problem. The clan $\sigma = 8$ and the tabu list $T = (2, 6)$ carry the additional information for the mating strategy.

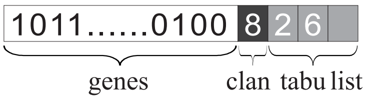


Fig. 2.1. Representation of chromosomes

2.3.2 Mating Strategy

In TMPGA, the mating of multiple parents is not unbridled but is restricted by the strategy of TS. The mating strategy of TMPGA is made up of two tactics in TS — the *tabu restriction* and the *aspiration criterion*. First, the clans in the tabu list are forbidden for the chromosome to mate with. A check on parents is carried out to see whether the mating of these parents is forbidden or not. Such a restrictive mechanism is helpful for maintaining population diversity through an incest-prevention manner [16, 37, 94]. Furthermore, TMPGA adopts the concept of polygamy: the check is only carried out on the relationship between the first parent and the other parents, which needs $2\nu(n-1)$ times of checks in the worst case. In contrast to the number $n\nu(n-1)$ of complete checks between all parents in the worst case, the manner of TMPGA largely reduces the computation cost in checking tabu and still preserves certain level of restriction. As shown in Fig. 2.2, the tabu restriction occurs when the first parent discovers that its clan exists in the tabu lists of other parents, and vice versa. Once the tabu occurs, the mating is judged to be invalid, unless the aspiration criterion is satisfied.

Definition 2.3 (Tabu). *Suppose we have n parents $\mathbf{c}_1, \dots, \mathbf{c}_n$. Let σ_i be the clan and T_i be the tabu list of the parent $\mathbf{c}_i$ with $i \in \{1, \dots, n\}$. The function* Tabu *in TMPGA is defined by*

$$\mathrm{Tabu}(\mathbf{c}_1, \mathbf{c}_2, \dots, \mathbf{c}_n) \overset{\mathrm{def}}{=} \begin{cases} True & \textit{if } \exists i \in \{2, \dots, n\} : \sigma_1 \in T_i \textit{ or } \sigma_i \in T_1, \\ False & \textit{otherwise.} \end{cases}$$

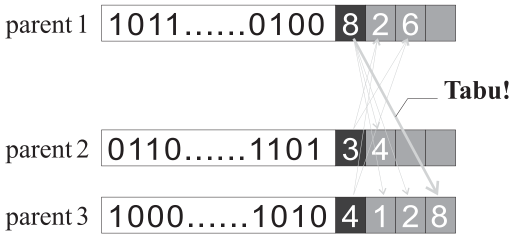


Fig. 2.2. Tabu checking procedure

The aspiration criterion defines that a mating is acceptable if the resultant offspring is superior to the best chromosome so far, even though this mating is tabu. This release from restriction provides moderate reinforcement in selection pressure. That is to say, the aspiration criterion is provided against the tabu restriction so as to balance diversification and intensification in search. The interaction of the tabu restriction and the aspiration criterion consequently achieves a harmony in diversity maintenance and selection pressure.

Definition 2.4 (Aspiration). *Suppose we have s offspring $\mathbf{c}'_1, \dots, \mathbf{c}'_s$. Let f be the fitness function for maximization and $\mathbf{c}$ be the best chromosome so far. The function* Aspiration *in TMPGA is defined by*

$$\mathrm{Aspiration}(\mathbf{c}'_1, \mathbf{c}'_2, \dots, \mathbf{c}'_s) \overset{\mathrm{def}}{=} \begin{cases} True & \textit{if } \exists i \in \{1, \dots, s\} : f(\mathbf{c}'_i) > f(\mathbf{c}), \\ False & \textit{otherwise.} \end{cases}$$

A mating is classified *invalid* and is not allowed, if the tabu restriction takes place and the aspiration criterion is not satisfied.

Definition 2.5 (Validity of Mating). *Suppose we have n parents $\mathbf{c}_1, \dots, \mathbf{c}_n$. Perform the reproduction with these n parents and we obtain the offspring $\mathbf{c}'_1, \dots, \mathbf{c}'_s$. The validity of mating is then defined by*

$$\mathrm{Mating} \overset{\mathrm{def}}{=} \begin{cases} Valid & \textit{if } (\mathrm{Tabu}(\mathbf{c}_1, \dots, \mathbf{c}_n) = False) \\ & \quad \textit{or } (\mathrm{Aspiration}(\mathbf{c}'_1, \dots, \mathbf{c}'_s) = True), \\ Invalid & \textit{otherwise.} \end{cases}$$

Note that only the offspring reproduced from a valid mating are allowed to be put into the subpopulation. After a valid mating, the parents who participated in it need to update their own tabu lists: The first parent adds all its mates' clans to its tabu list, while the other parents only add the clan of the first parent to their tabu lists. In addition, the offspring will inherit the updated tabu list from one of their parents. The operation of updating the tabu list works in a FIFO (first-in-first-out) manner. The oldest clans will be released from the tabu list, as the tabu list is full and new forbidden clans are added to it. These released clans regain the permission to be mated with. Figure 2.3 illustrates the procedure for updating the tabu list. The first parent adds the clans (3, 4) of the mates to its tabu list; other parents only add the clan (8) of the first parent to their own tabu lists. The released clan (6) means that the first parent regains the permission to mate with the chromosomes belonging to clan (6).

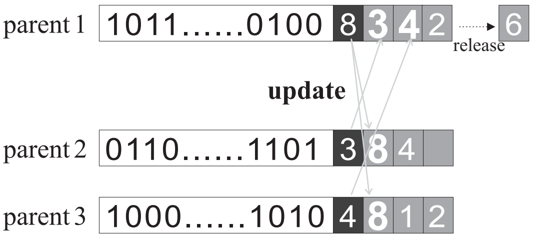


Fig. 2.3. Update of tabu list

According to the validity of mating, TMPGA controls the reproduction and the number of parents in the following way: If a mating is valid, its offspring will be put into the subsequent subpopulation. Otherwise, TMPGA will remove all the tabu parents to enable this mating valid and then perform crossover with the remaining parents to reproduce offspring again. In case that all mates of the first parent are removed, the selection operator will randomly choose a new mate from the population to keep the mating having at least two parents. The number of parents, therefore, varies with the situation of the population. In the beginning of evolution, the number of parents is relatively large because the tabu restriction is less likely to occur and the aspiration criterion is easier to meet, compared to the later phase of evolution. This large number of parents leads to a more diverse search around the problem space [36]; that is, it enhances the exploration. In the course of evolution, the probability of invalid mating will become higher and higher owing to the loss of population diversity. The number of parents will then decrease in response to the increasing rate of invalid mating. Finally, the number of parents will be reduced to two — at that time, multi-parent crossover will degenerate into 2-parent crossover. In view of diagonal crossover, this decline of parents keeps the search from violent disruptiveness when the search is approaching the promising region. On the whole, the mating strategy adjusts the number of parents under consideration of both exploration and exploitation in different phases of evolution.

2.3.3 Algorithm

The proposed algorithm TMPGA is presented in Algorithm 3. Most procedures of TMPGA follow the original MPGA except the screening process of tabu strategy. An additional rule is proposed to reduce the computation cost of tabu checking. This rule initially sets the number n to the predetermined maximal value. At each generation the number n is changed to the largest number of parents among all valid mating in the previous generation. Using the rule, the number of parents is adjusted according to the situation of population in the preceding generation. Generally it will decrease the number of parents participating in a mating. As the number is reduced to two, the checks of tabu restriction and aspiration criterion will be omitted and the multi-parent crossover will degenerate into a 2-parent crossover. The reason to omit the check is that the population at that time has lost most of its diversity and will yield an extremely high probability of invalid mating — it implies that most of the mating will be invalid when the trial number of parents is reduced to two. In this case, performing checks will be trivial and will pay the expensive cost of computation in the check and the removal of tabu parents. Omitting the check procedure as $n = 2$, therefore, can substantially reduce the number of trivial checks and then the computation cost.

Algorithm 3 TMPGA

```
$t \leftarrow 0$
*Initialize* population $C(t)$
*Evaluate* $C(t)$
$n =$MAX_PARENTS
**repeat**
  $n^* \leftarrow 2$
  **repeat**
    $\mathbf{x}_1, \ldots, \mathbf{x}_n \leftarrow Select(C(t))$
    $\mathbf{y}_1, \ldots, \mathbf{y}_n \leftarrow Reproduce(\mathbf{x}_1, \ldots, \mathbf{x}_n)$
    **if** $((n > 2)$ AND $(Tabu(\mathbf{x}_1, \ldots, \mathbf{x}_n)$
        AND (NOT $Aspiration(\mathbf{y}_1, \ldots, \mathbf{y}_n)))$ **then**
      $\mathbf{x}_1, \ldots, \mathbf{x}_{n'} \leftarrow RemoveTabu(\mathbf{x}_1, \ldots, \mathbf{x}_n)$
      $\mathbf{z}_1, \ldots, \mathbf{z}_{n'} \leftarrow Reproduce(\mathbf{x}_1, \ldots, \mathbf{x}_{n'})$
      $C(t+1) \leftarrow C(t+1) \cup \{\mathbf{z}_1, \ldots, \mathbf{z}_{n'}\}$
      $n^* \leftarrow \max(n^*, n')$
    **else**
      $C(t+1) \leftarrow C(t+1) \cup \{\mathbf{y}_1, \ldots, \mathbf{y}_n\}$
      $n^* \leftarrow \max(n^*, n)$
  **until** $(C(t+1)$ is filled)
  $C(t+1) \leftarrow Survivor(C(t), C(t+1))$
  $n \leftarrow n^*$
  $t \leftarrow t+1$
**until** (termination criterion is satisfied)
```

2.4 Experimental Results

Multi-parent crossovers can be classified into 'n-parents-n-children' crossovers (e.g. diagonal crossover) and 'n-parents-1-child' crossovers (e.g. scanning crossover) according to the number of offspring reproduced by every performance of crossover. This number of offspring, however, has an effect on the likelihood of aspiration and further the validity of mating. As aforementioned, this validity on the one hand sifts the parents and, on the other hand, controls the number of parents. In the following sections we will examine the effectiveness of TMPGA on these two sorts of crossovers. In terms of n-parents-n-children crossovers, we adopt diagonal crossover in the experiments. For n-parents-1-child crossover, two scanning crossovers (U-Scan and OB-Scan) will be examined. Since an n-parents-n-children crossover can be used as an n-parents-1-child crossover by randomly picking one child from n children, we additionally experiment with diagonal crossover for MPGAs using n-parents-1-child crossover.

2.4.1 MPGAs Using n-parents-n-children Crossover

In this section we focus on diagonal crossover, an n-parents-n-children crossover. Four common test functions [8, 20, 33] are adopted in our experiments: De Jong's second test function (F2), the Rastrigin (RAS), the Schwefel (SCH), and the Griewangk (GRI). Table 2.1 describes these test functions[1] and the related parameters used in our experiments. The simple GA (using two parents) and MPGA (using more than two parents) are additionally implemented for performance comparison with TMPGA.

Table 2.1. Test functions

Function	N	Bits of x_i	l
$f_{\mathrm{F2}} = 100\left(x_2 - x_1^2\right)^2 + \left(x_1 - 1\right)^2, x_i \in [-2.048, 2.047]$	2	12	24
$f_{\mathrm{RAS}} = 10N + \sum\limits_{i=1}^{N}\left[x_i^2 - 10\cos\left(2\pi x_i\right)\right], x_i \in [-5.12, 5.11]$	20	10	200
$f_{\mathrm{SCH}} = 418.98291N + \sum\limits_{i=1}^{N} -x\sin\left(\sqrt{\lvert x_i\rvert}\right), x_i \in [-512, 511]$	10	10	100
$f_{\mathrm{GRI}} = 1 + \sum\limits_{i=1}^{N}\frac{x_i^2}{4000} - \prod\limits_{i=1}^{N}\cos\left(\frac{x_i}{\sqrt{i}}\right), x_i \in [-512, 511]$	10	10	100

The setting of GA is listed in Table 2.2. This setting is also applied to MPGA and TMPGA. Each experiment setting includes 50 independent runs. The crossover for MPGA and TMPGA is diagonal crossover and that for GA is 2-parent diagonal crossover, i.e. one-point crossover. Here we only show the results of the MPGA using 15 parents because it performs best among our experiments with 11 to 15 parents[2].

[1] A more detailed description of these functions is given in Appendix A.

[2] These numbers of parents were adopted and showed superior success rates in the original work of diagonal crossover [36].

The number of parents for TMPGA follows MPGA by setting 15 parents initially; this number will be adjusted adaptively by the strategy of TS afterwards. The size of tabu list in TMPGA is empirically set to 10 for a moderate restriction.

Table 2.2. The setting of GA in experiments

GA type	Generational GA
Representation	Bit string
Population size	200
Selection	Roulette wheel selection
Crossover	Diagonal crossover
Crossover rate	1.0
Mutation	Bit-flip mutation
Mutation rate	$\frac{1}{l}$
Termination	100 generations

Figure 2.4 compares the convergence of TMPGA with those of GA and MPGA on the four test functions. The results show that TMPGA converges faster than both GA and MPGA on all test problems, except the first half of convergence on F2. In addition, MPGA converges faster than GA on SCH and GRI, but converges slower than GA on F2 and RAS. In terms of solution quality, TMPGA yields better solutions than GA does on all functions and than MPGA does on RAS. The differences of the best solutions obtained from MPGA and TMPGA are insignificant on F2, SCH, and GRI.

The above comparison concerning the number of generations is not so fair since TMPGA spends additional computation on tabu checking and trivial mating in each generation. For this, we further compare these algorithms with respect to running time. The comparing algorithms are coded in C language and run on an Intel Pentium III – 1.7GHz machine. Figure 2.5 demonstrates that the additional computation of TMPGA slightly depresses its superiority, compared to the results in Fig. 2.4. However, TMPGA still achieves faster convergence than GA on RAS, SCH, and GRI. The algorithm TMPGA also converges faster than MPGA does on all test functions, while the convergence of TMPGA and that of MPGA are close on SCH and GRI. The superiority of TMPGA supports that the extra computation cost of tabu checking is worthwhile in terms of convergence speed.

Table 2.3 further quantifies the improvements of MPGA and TMPGA in the convergence speed of GA. These results indicate the number of generations as well as the time taken for MPGA and TMPGA to achieve GA's best fitness in our experiments. Table 2.3 shows that TMPGA converges faster than GA by 24 to 66 generations or by 0.881 to 3.352 seconds; namely, TMPGA saves 22% to 69% of GA's convergence time. MPGA also converges faster than GA except on RAS, where MPGA performs similarly to GA. The presented performance comparison validates the superiority of TMPGA over GA and MPGA in convergence speed.

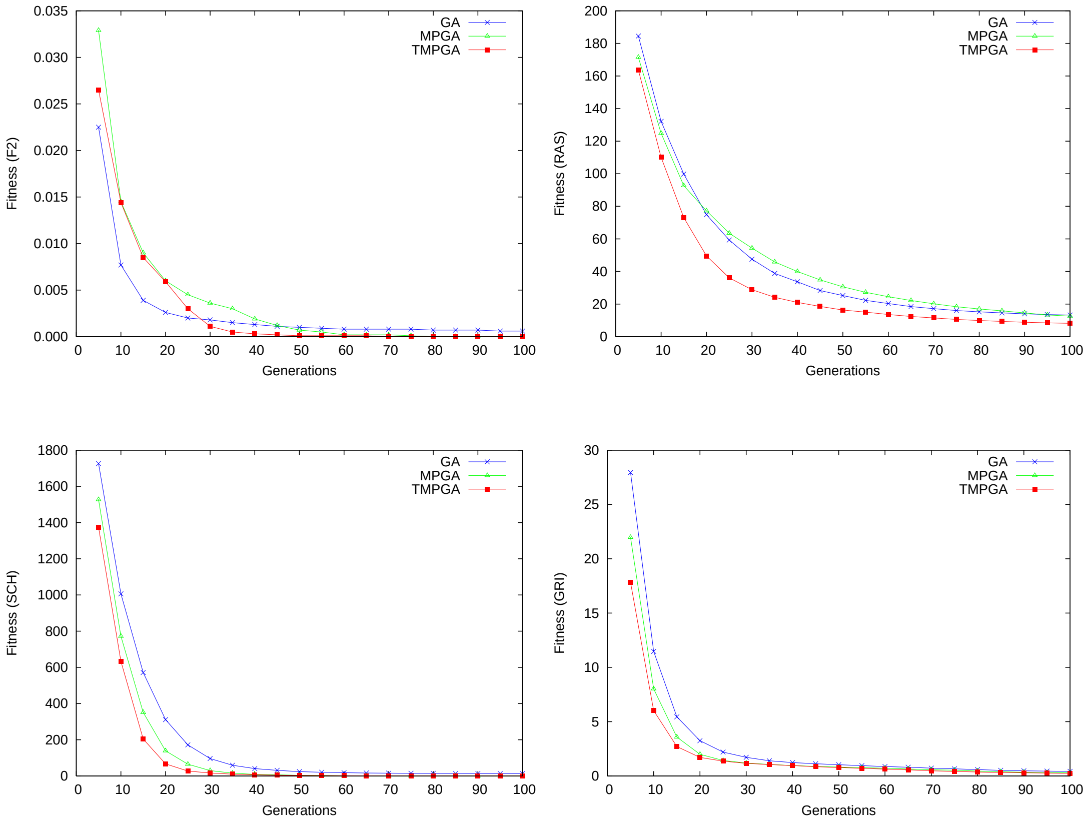


Fig. 2.4. Performance comparison of GA, MPGA, and TMPGA in terms of generations

Table 2.3. Comparison of generations and time for MPGA and TMPGA to achieve the best fitness of GA

	Generations			Time (sec)		
	GA	MPGA	TMPGA	GA	MPGA	TMPGA
F2	96	53	35	4.840	2.036	1.488
RAS	100	97	62	4.398	4.277	3.041
SCH	199	38	34	4.207	1.478	1.470
GRI	100	84	76	4.009	3.289	3.128

We further investigate the impact of the tabu restriction and the aspiration criterion on TMPGA's performance and the number of parents. The number of tabu events, as shown in Fig. 2.6, peaks around 40 generations on F2 and around 20 to 25 generations on RAS, SCH, and GRI. Compared to the convergence of TMPGA in Fig. 2.4, the peak of tabu events corresponds to the turning point of convergence. We attribute this correspondence to the exhaustion of population diversity: In the course of evolution the population will inevitably lose its diversity if the mutation rate is not comparable to the rate of genetic drift. The resulting similarity of chromosomes in the population is

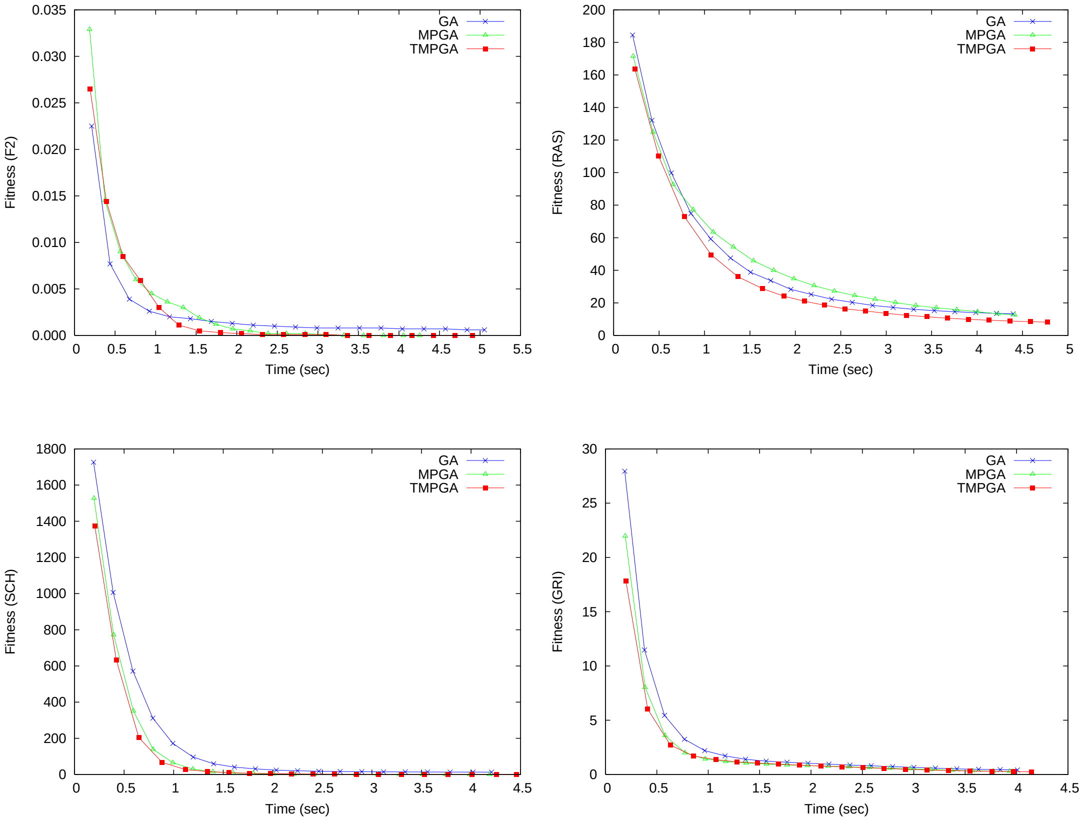


Fig. 2.5. Performance comparison of GA, MPGA, and TMPGA in terms of running time

reflected in the high probability of tabu events since the population is full of chromosomes with the same clans. The peak of tabu events implies that the population is too similar for the search to explore the search space effectively. Therefore, the convergence of TMPGA slows down after the peak of tabu events.

Figure 2.7 depicts the variation in the number of parents on the four test functions. Clearly, the number of parents decreases from 15 to 2 in the course of evolution. As mentioned in Section 2.3, the validity of mating affects the number of parents in TMPGA. Considering the growing number of tabu events and the shrinking number of aspiration events shown in Fig. 2.6, the mating becomes more likely to be invalid. As a result of the increasing number of invalid mating, the number of parents decreases with generations, which contributes to restrain the disruptiveness caused by the increase of parents in diagonal crossover. It is noteworthy that the influence of the tabu strategy on the number of parents is two-way: The tabu restriction reduces the number of parents, while the reduced number of parents will further lower the probability to induce the tabu restriction in the next generation. All in all, the synergy of the tabu restriction and the aspiration criterion achieves not only a harmonious mating but an adaptive

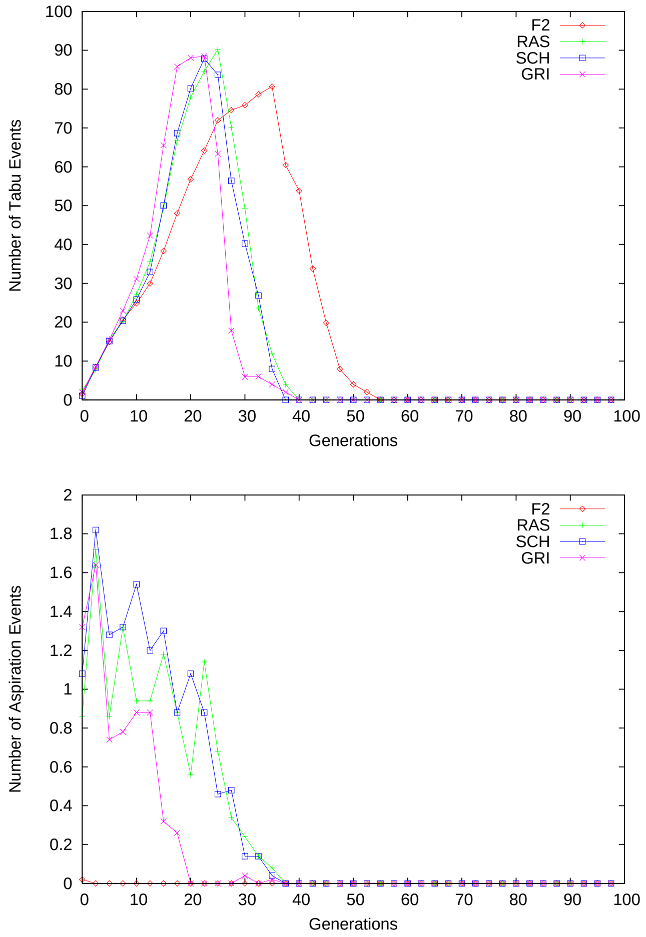


Fig. 2.6. Variation in the numbers of tabu events (top) and aspiration events (bottom) averaged over 50 runs of TMPGA on the four test functions

restraint on disruptiveness; consequently, it leads to the superiority of TMPGA over GA and MPGA.

2.4.2 MPGAs Using *n*-parents-1-child Crossover

In this section we consider diagonal crossover and two scanning crossovers: U-Scan and OB-Scan. The test suite used in the previous section is also adopted here. We additionally extend the F2 function to 10 variables as the F2e function. Table 2.4 presents these test functions and the related parameters. The GA and MPGA are implemented for performance comparison with TMPGA. The setting of GA, MPGA, and TMPGA is listed in Table 2.5. Each experiment setting includes 100 independent runs. The crossover for GA is 2-parent diagonal crossover (i.e. one-point crossover) and 2-parent scanning crossover (i.e. uniform crossover). The number of parents is set to

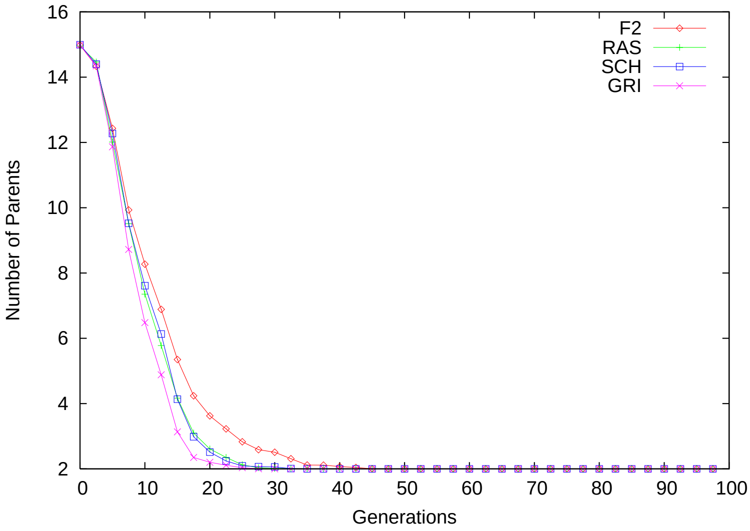


Fig. 2.7. Variation in the number of parents averaged over 50 runs of TMPGA on the four test functions

6 and 15 for MPGA and TMPGA, where this number in TMPGA will be adjusted adaptively during evolution. The size of tabu list in TMPGA is set to 6 for a looser restriction in response to the smaller population than that used in the previous section.

Table 2.4. Test functions

Function	N	Bits of x_i	l
$f_{\text{F2}} = 100\left(x_2 - x_1^2\right)^2 + \left(x_1 - 1\right)^2, x_i \in [-2.048, 2.047]$	2	12	24
$f_{\text{F2e}} = \sum_{i=1}^{N-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2\right], x_i \in [-2.048, 2.047]$	10	12	120
$f_{\text{RAS}} = 10N + \sum_{i=1}^{N} \left[x_i^2 - 10\cos\left(2\pi x_i\right)\right], x_i \in [-5.12, 5.11]$	10	10	100
$f_{\text{SCH}} = 418.98291N + \sum_{i=1}^{N} -x\sin\left(\sqrt{\lvert x_i \rvert}\right), x_i \in [-512, 511]$	10	10	100
$f_{\text{GRI}} = 1 + \sum_{i=1}^{N} \frac{x_i^2}{4000} - \prod_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right), x_i \in [-512, 511]$	10	10	100

Figure 2.8 compares the convergence of TMPGA with those of GA and MPGA using diagonal crossover, U-Scan, and OB-Scan with respect to 15 and 6 parents on SCH. More performance comparisons on F2, RAS, GRI, and F2e are presented in Figs. B.1–B.3. These figures show that TMPGA using diagonal crossover outperforms GA and MPGA using diagonal crossover. In the case of 6-parent diagonal crossover, the improvement of TMPGA over MPGA is more significant. As for U-Scan, there exists little difference in the convergence of these three algorithms, regardless of the number of parents adopted in U-Scan. Moreover, these figures reveal that the MPGA

Table 2.5. The setting of GA in experiments

GA type	Generational GA
Representation	Bit string
Population size	128
Selection	Linear ranking ($\eta^+ = 1.5$)
Crossover	Diagonal crossover, U-Scan, and OB-Scan
Crossover rate	1.0
Mutation	Bit-flip mutation
Mutation rate	$\frac{1}{l}$ and 0.1
Termination	500 generations

using OB-Scan with mutation rate $\frac{1}{l}$ performs poorly in comparison with GA, and yet TMPGA can improve the poor performance of MPGA, especially in the solution quality. However, the performance of TMPGA using OB-Scan is still inferior to GA using uniform crossover.

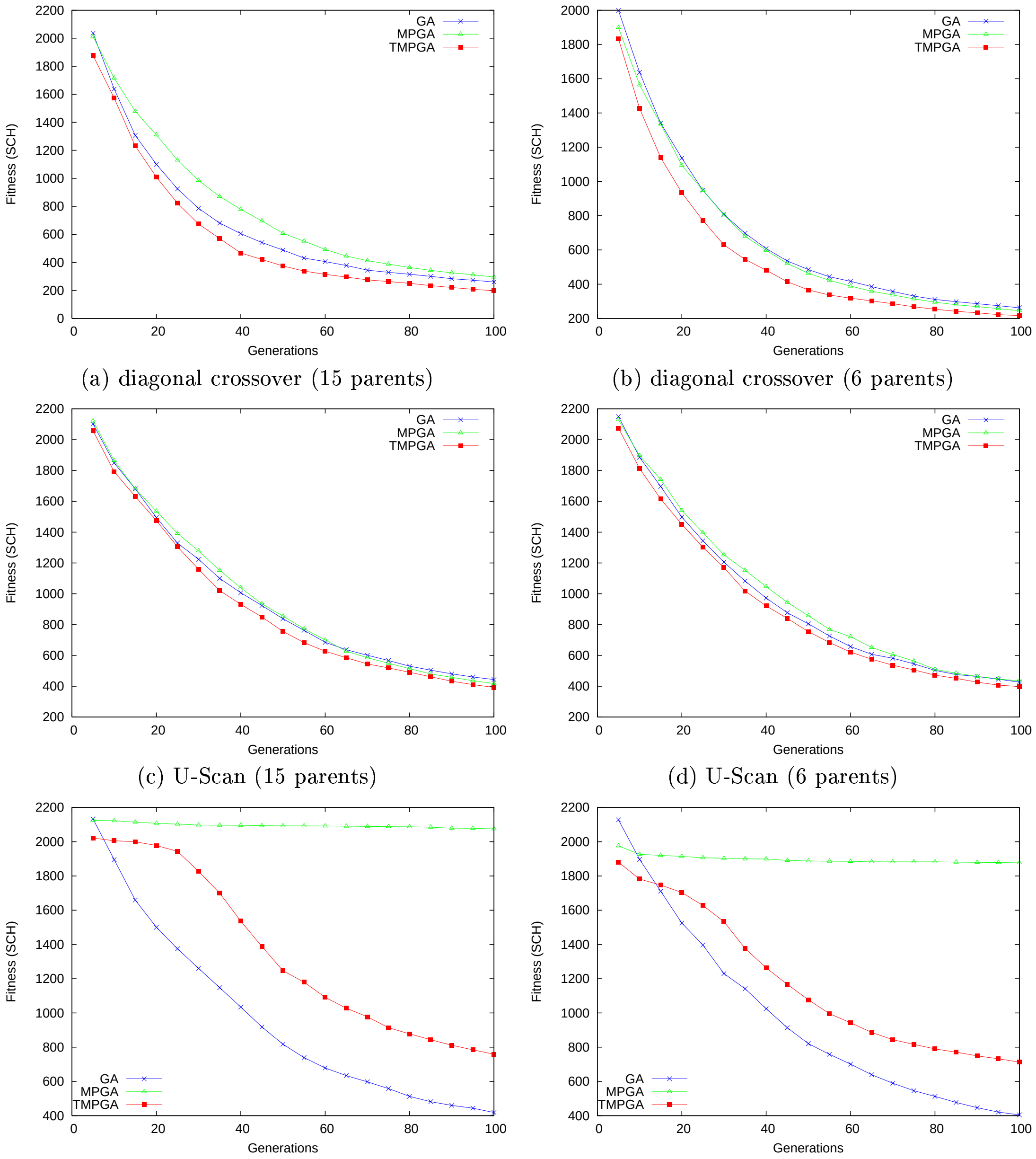
We further conduct experiments with a high mutation rate 0.1 and show the results on SCH in Fig. 2.9. More experimental results on other test functions are presented in Figs. B.4–B.6. Basically, the comparative outcomes of GA, MPGA, and TMPGA with mutation rate 0.1 are similar to those with mutation rate $\frac{1}{l}$, but the superiority of TMPGA over GA and MPGA becomes more significant in mutation rate 0.1. By contrast to mutation rate $\frac{1}{l}$, MPGA using OB-Scan with mutation rate 0.1 performs comparably and even better than GA. Furthermore, TMPGA can reinforce this improvement and far outperform MPGA and GA in the use of OB-Scan.

2.5 Summary

This chapter presented the tabu multi-parent genetic algorithm (TMPGA) to deal with two issues in MPGAs: the mating and the number of parents. The algorithm TMPGA integrates the tactics of TS into the mating of MPGA. An additional memory structure consisting of the clan number and the tabu list is appended to the original structure of chromosomes. This memory structure records the trajectory of evolution and serves as the basis of the mating strategy. First, the tabu list forbids certain chromosomes from mating so that the population diversity can be maintained in an incest-prevention way. Second, the aspiration criterion provides a possibility to override the tabu restriction in order to supply selection pressure. As a result of the synergy of the tabu restriction and the aspiration criterion, TMPGA controls the mating of more than two parents with a good balance between exploration and exploitation. The validity of mating, furthermore, adjusts the number of parents adaptively.

Several experiments are conducted to verify the effectiveness of TMPGA. Experimental results show that TMPGA outperforms GA and MPGA in convergence speed and solution quality on four common test functions. Furthermore, we investigated the influence of tabu and aspiration on the convergence and the variation of the number of parents in TMPGA. The results indicate that the peak of tabu events corresponds to the turning point when the convergence slows down. In addition, the growing number of tabu events and the shrinking number of aspiration events lead to the decrease of parents. This reduced number of parents, on the other hand, will lower the probability to induce the tabu event. Thus the influence of tabu restriction on the number of parents is two-way.

The above preferable performance of TMPGA validates the capability of the proposed mating strategy. However, these consequences are empirical. In addition, even though TMPGA concerns the variation in the number of parents and its correlation with tabu restriction, it provides little information about the general impact of raising parents on the performance of multi-parent crossover and the whole MPGA. In the following chapters, we will conduct theoretical analysis to explore the role of the number of parents in the performance of MPGAs.



(a) diagonal crossover (15 parents)

(b) diagonal crossover (6 parents)

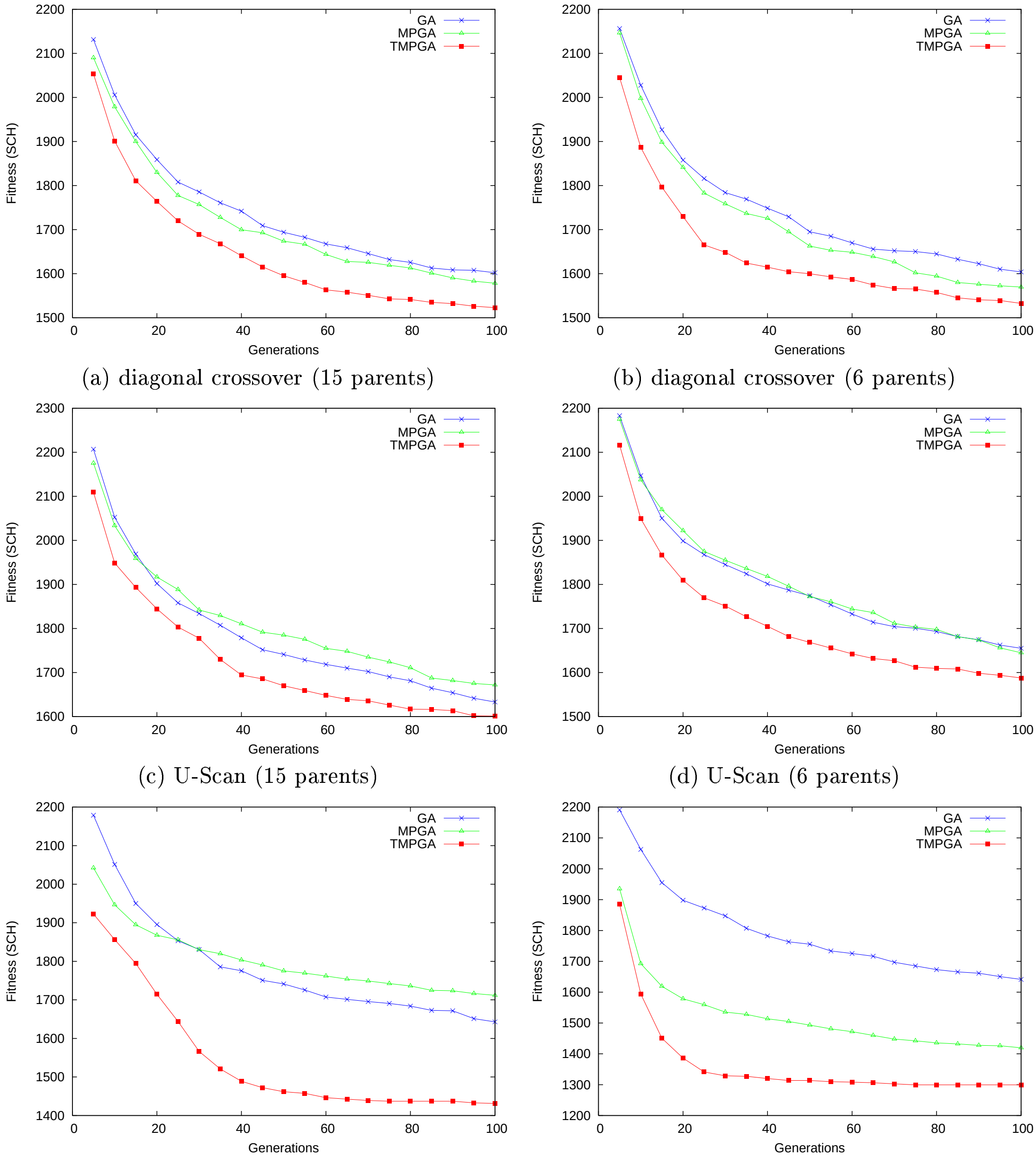
(c) U-Scan (15 parents)

(d) U-Scan (6 parents)

(e) OB-Scan (15 parents)

(f) OB-Scan (6 parents)

Fig. 2.8. Performance comparison of GA, MPGA, and TMPGA using diagonal crossover, U-Scan, and OB-Scan with mutation rate 0.01



(a) diagonal crossover (15 parents)

(b) diagonal crossover (6 parents)

(c) U-Scan (15 parents)

(d) U-Scan (6 parents)

(e) OB-Scan (15 parents)

(f) OB-Scan (6 parents)

Fig. 2.9. Performance comparison of GA, MPGA, and TMPGA using diagonal crossover, U-Scan, and OB-Scan with mutation rate 0.1

3

Analysis Based on Uniform Population Model

This chapter addresses the following questions through an analysis based on a specific population model:

- When will MPGAs outperform GAs?
- Why do MPGAs perform better?

The procedure for analysis, as illustrated in Fig. 3.1, focuses on one generation rather than the whole evolution (cf. Fig. 1.1). First, a simplified population model, called *uniform population*, is presented as a systematic population environment for analysis. Second, a criterion based on uniform population is presented to evaluate the performance of crossover. Accordingly, we analyze two multi-parent crossovers: *uniform scanning crossover* (U-Scan) and *occurrence based scanning crossover* (OB-Scan). The analysis reveals the ineffectiveness of adopting more than two parents in U-Scan; moreover, it proves that increasing the number of parents will intensify the probability for OB-Scan to reproduce better or worse chromosomes. The experimental results on four test functions show a high level of consistence with the analytical claims, which validates the capability of this analysis.

The present chapter extends the work [101].

3.1 Introduction

A number of theoretical analyses on MPGAs have been proposed. Nevertheless, as concluded in Chapter 1, the influence of raising parents on the performance of crossover is still left open. For that matter, we conduct a theoretical analysis on the effect of applying more parents in two multi-parent crossovers: U-Scan and OB-Scan.

The analysis presented in this chapter is based on an assumption about the population: the *uniform population* model. The chromosomes in uniform population are defined to have the same hamming distance from the unique optimal solution. The composition of population, under the assumption of uniform population, is simplified systematically and then becomes tractable for analysis. A relevant criterion is further presented to evaluate the performance of crossover. The analysis gives the probability

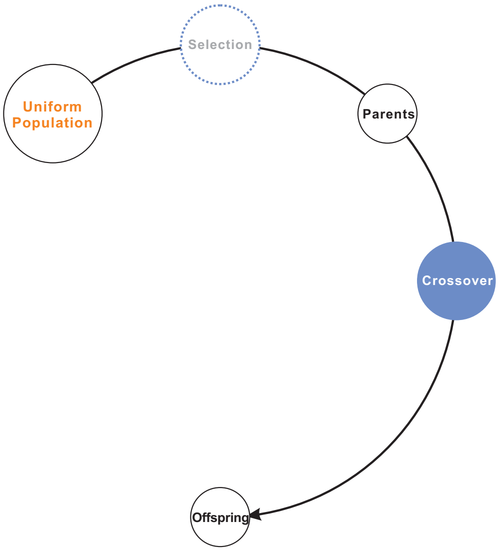


Fig. 3.1. Procedure for analysis based on uniform population

for U-Scan and OB-Scan to reproduce better, equal, or worse offspring. In addition, a conjecture about the correlation between the number of parents and the level of exploitation in OB-Scan is proposed according to the analytical results. These claims are further examined by a series of experiments on four common test functions.

The rest of the present chapter is organized as follows. In Section 3.2 we present the uniform population model. Section 3.3 elaborates on the theoretical analysis of U-Scan and OB-Scan. Section 3.4 presents the experimental validation. Finally, conclusions are drawn in Section 3.5.

3.2 Uniform Population

In this chapter, we present the uniform population model as the basis for analyzing the effectiveness of crossover. Crossover is an operator used to recombine and exchange the genetic material of parents. The outcome of crossover, nevertheless, depends upon the composition of population and the selection process. To overcome the difficulty arising from these two factors in analysis, we propose a systematic population model, called *uniform population*, to simplify the conditions of population. Based on uniform population, a criterion for evaluating the performance of crossover will be presented in the next section.

Assume that there exists one unique optimal solution for the given problem. Uniform population is defined as the population in which each chromosome differs from the optimal solution by k genes. In other words, the hamming distance between the optimal solution and each chromosome in uniform population is exactly k. Here the value k is called the *order* of uniform population. For example, without loss of generality we assume the optimal solution to be '000...000'. A 3-order uniform population consisting m chromosomes is illustrated in Fig. 3.2.

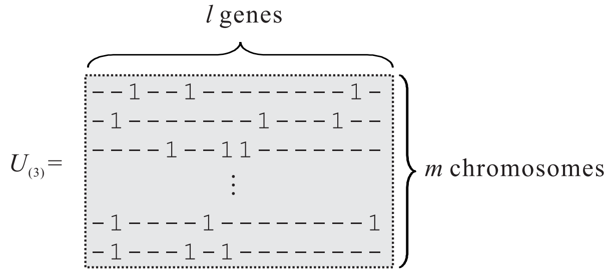


Fig. 3.2. A *3*-order uniform population ('1' represents the bit 1, and '-' represents the bit 0)

Definition 3.1 (Distinct Genes).

1. *For a chromosome* **c***, the* distinct genes *of* **c** *are the different genes between the chromosome* **c** *and the optimal solution* $\mathbf{c}^*$*.*
2. *Let* $D(\mathbf{c})$ *be the set of distinct genes in* **c** *and* $|D(\mathbf{c})|$ *be the cardinality of* $D(\mathbf{c})$*. We have*
$$|D(\mathbf{c})| = H(\mathbf{c}, \mathbf{c}^*),$$
where $H : \{0,1\}^l \times \{0,1\}^l \to \{0,1,\ldots,l\}$ *is the hamming distance function.*

Definition 3.2 (k-positive, k-ineffective, and k-negative). *Given* $k \in \mathbb{N}$*, we classify a chromosome* **c** *as*

$$\mathbf{c} \textit{ is } \begin{cases} k\textit{-positive} & \textit{if } |D(\mathbf{c})| < k, \\ k\textit{-ineffective} & \textit{if } |D(\mathbf{c})| = k, \\ k\textit{-negative} & \textit{if } |D(\mathbf{c})| > k. \end{cases}$$

Definition 3.3 (k-order Uniform Population). *A* k-order uniform population*, denoted as* $U_{(k)}$*, is defined by*

$$U_{(k)} \overset{\text{def}}{=} \left\{\mathbf{c} \mid \mathbf{c} \in \{0,1\}^l \text{ and } |D(\mathbf{c})| = k\right\}.$$

3.3 Analysis Based on k-order Uniform Population

This section analyzes the influence of using more parents upon the performance of U-Scan and OB-Scan. As aforementioned, crossover is dependent upon the composition of the population and the selection process. In the previous section we have proposed uniform population as the composition of population. Here we additionally assume the selection process is a random selection. Under these two assumptions, we embark on the analysis of the influence of increasing parents in crossover.

First, we define the criterion for evaluating the performance of crossover.

Definition 3.4. *Let $\mathbf{c}'$ be the offspring reproduced by a crossover $\mathcal{X}$ with n parents selected from $U_{(k)}$. The symbols p^+, $p^=$, and p^- are defined by the probability that $\mathbf{c}'$ is k-positive, k-ineffective, and k-negative, respectively. For example, the probability p^+ is defined by*

$$\begin{aligned} p^+(\mathcal{X}, n, U_{(k)}) &\overset{\mathrm{def}}{=} \Pr\{\mathbf{c}' \textit{ is } k\textit{-positive} \mid \mathbf{c}' = \mathcal{X}(\mathbf{c}_1, \ldots, \mathbf{c}_n \mid \mathbf{c}_i \in U_{(k)})\} \\ &= \Pr\{|D(\mathcal{X}(\mathbf{c}_1, \ldots, \mathbf{c}_n))| < k \mid \mathbf{c}_i \in U_{(k)}\} \end{aligned}$$

Remark 3.5. In scanning crossover, the sequence of parents has no influence on the probability p^+, $p^=$, and p^-.

Definition 3.6 (Performance of Crossover). *The performance $\mathcal{F}$ of crossover $\mathcal{X}$ with n parents selected from $U_{(k)}$ is defined by*

$$\begin{aligned} \mathcal{F}(\mathcal{X}, n, U_{(k)}) \overset{\mathrm{def}}{=} (\ & p^+(\mathcal{X}, n, U_{(k)}), \\ & p^=(\mathcal{X}, n, U_{(k)}), \\ & p^-(\mathcal{X}, n, U_{(k)})\ ). \end{aligned}$$

The equivalence[1] *of performance is further defined by*

$$\begin{aligned} & \mathcal{F}(\mathcal{X}, \mu, U_{(k)}) = \mathcal{F}(\mathcal{X}, \nu, U_{(k)}) \\ \Longleftrightarrow\ & (p^+_\mu = p^+_\nu) \text{ and } (p^=_\mu = p^=_\nu) \text{ and } (p^-_\mu = p^-_\nu)\ , \end{aligned} \tag{3.1}$$

where p^+_μ denotes $p^+(\mathcal{X}, \mu, U_{(k)})$, and so forth[2].

Next, we investigate the performance of scanning crossover by calculating the expected number of distinct genes in the offspring. To this end, we need the probability p that the distinct gene occurs at a parental locus. Based on this probability p, we can derive the probability that there exist x of n given parents possessing the distinct gene at some locus. In the following analysis we will make use of *binomial distribution* [79]. The probability mass function (p.m.f) $B(x; n, p)$ of binomial distribution $B(n, p)$ is

$$B(x; n, p) = \binom{n}{x} p^x (1-p)^{n-x}, \tag{3.2}$$

where $n, x \in \mathbb{Z}_*$. It holds for the expectation of a binomial random variable X that

$$\mathrm{E}[X] = \sum_{x=0}^{n} x \cdot B(x; n, p) = np. \tag{3.3}$$

[1] Here we only discuss the equivalence of performance. However, the correlation between (p^+, $p^=$,p^-) and the performance of MPGAs is an issue worthy of future study.

[2] In fact, any two conditions in (3.1) suffice the equivalence of performance since $p^+ + p^= + p^- = 1$.

Lemma 3.7. *Given a chromosome* $\mathbf{c}$ *selected from* $U_{(k)}$*, the probability that a gene* c_i *with* $i \in \{1, \ldots, l\}$ *is a distinct gene is* $p = \frac{k}{l}$*.*

Proof. The definition of k-order uniform population tells that each chromosome in this population has exact k distinct genes. Therefore, for a chromosome selected from $U_{(k)}$ the probability that c_i is a distinct gene is

$$p = \Pr\{c_i \text{ is a distinct gene}\} = \frac{k}{l}.$$

□

Lemma 3.8. *Given* n *parents randomly selected from* $U_{(k)}$*, the probability that* x *of* n *parents hold the distinct gene at locus* $i \in \{1, \ldots, l\}$ *is* $B(x; n, p)$*.*

Proof. According to Lemma 3.7, the probability for a chromosome selected from $U_{(k)}$ to possess the distinct gene at locus i is $p = \frac{k}{l}$. Since the random selection is a Bernoulli process, the number of selected chromosomes having the distinct gene at locus i holds a binomial distribution $B(n, p)$. Hence the probability that x of n chromosomes possess distinct genes at some locus is $B(x; n, p)$. □

3.3.1 Analysis of U-Scan

In the operation of U-Scan, offspring inherit a gene randomly from one of the parents for each locus. This random manner implies the probability is equal for each parent to give its genes. Additionally, Lemma 3.8 showed the probability that x of n parents possess the distinct gene at some locus. According to these properties, we can derive the probability that an offspring locus is assigned with the distinct gene. Furthermore, the expected number of distinct genes in the offspring can be derived.

Lemma 3.9. *Given* n *parents selected from* $U_{(k)}$*, the probability* p_{u} *for U-Scan* $\mathcal{X}_{\mathrm{u}}$ *to yield a distinct gene for a locus is*

$$p_{\mathrm{u}} = \frac{k}{l}.$$

Proof. Let $\mathbf{c}'$ be the offspring reproduced by $\mathcal{X}_{\mathrm{u}}$. Assume x of n selected parents possess the distinct gene at locus $i \in \{1, \ldots, l\}$. Since U-Scan chooses the parent to give genes randomly, the probability for each parent to be chosen is $1/n$. This gives the probability for the offspring to inherit the distinct gene

$$\Pr\left\{c'_i \text{ is a distinct gene} \mid x\right\} = \frac{x}{n}.$$

Using the p.m.f. of x in Lemma 3.8 and (3.3), we have the probability

$$\begin{aligned} p_{\mathrm{u}} &= \sum_{x=0}^{n} \Pr\left\{c'_i \text{ is a distinct gene} \mid x\right\} \cdot \Pr(x) \\ &= \sum_{x=0}^{n} \left(\frac{x}{n}\right) \cdot B\left(x; n, p\right) = \frac{1}{n} \cdot np = p = \frac{k}{l}. \end{aligned}$$

□

Theorem 3.10 (Performance of U-Scan). *Using k-order uniform population $U_{(k)}$, the number of parents has no effect on the performance of U-Scan. That is,*

$$\mathcal{F}(\mathcal{X}_{\mathrm{u}}, n, U_{(k)}) = \mathcal{F}(\mathcal{X}_{\mathrm{u}}, n', U_{(k)}) \quad \text{for } n, n' \in \mathbb{N}_{>1}.$$

Proof. Let $\mathbf{c}'$ be the offspring reproduced by $\mathcal{X}_{\mathrm{u}}$ and $|D(\mathbf{c}')|$ be the number of distinct genes in $\mathbf{c}'$. Lemma 3.9 gives the probability p_{u} that U-Scan assigns the distinct gene to an offspring locus. In addition, scanning crossover processes each gene of an offspring independently. Therefore, processing l genes in this way, U-Scan has the probability that δ distinct genes occur in the offspring $\mathbf{c}'$:

$$\Pr\left\{\left|D(\mathbf{c}')\right| = \delta\right\} = \binom{l}{\delta} (p_{\mathrm{u}})^{\delta} (1 - p_{\mathrm{u}})^{l-\delta} = B\left(\delta; l, p_{\mathrm{u}}\right). \tag{3.4}$$

This gives the probability p^{+}, $p^{=}$, and p^{-} of U-Scan $\mathcal{X}_{\mathrm{u}}$, respectively:

$$p^{+}(\mathcal{X}_{\mathrm{u}}, n, U_{(k)}) = \Pr\left\{\left|D(\mathbf{c}')\right| < k\right\} = \sum_{\delta=0}^{k-1} B\left(\delta; l, p_{\mathrm{u}}\right) \tag{3.5}$$

$$p^{=}(\mathcal{X}_{\mathrm{u}}, n, U_{(k)}) = \Pr\left\{\left|D(\mathbf{c}')\right| = k\right\} = B\left(k; l, p_{\mathrm{u}}\right) \tag{3.6}$$

$$p^{-}(\mathcal{X}_{\mathrm{u}}, n, U_{(k)}) = \Pr\left\{\left|D(\mathbf{c}')\right| > k\right\} = \sum_{\delta=k+1}^{l} B\left(\delta; l, p_{\mathrm{u}}\right) \tag{3.7}$$

Clearly, the above probabilities are independent of the value n as p_u is independent of n. This implies that for $n, n' \in \mathbb{N}_{>1}$

$$\mathcal{F}(\mathcal{X}_{\mathrm{u}}, n, U_{(k)}) = \mathcal{F}(\mathcal{X}_{\mathrm{u}}, n', U_{(k)}).$$

□

Figure 3.3 plots the performance $(p^{+}, p^{=}, p^{-})$ of U-Scan in k-order uniform population by (3.5)–(3.7). Here chromosomes are assumed to be encoded as 100-bit strings and the order k ranges from 1 to $l - 1$. Figure 3.3 only shows 2-parent U-Scan because Theorem 3.10 has proved that the performance of U-Scan is independent of the number n. This figure indicates that the probabilities for U-Scan to reproduce k-positive and k-negative offspring are around 0.44; in addition, both profiles are smooth in terms of the order k. This phenomenon implies U-Scan and its degenerated version, i.e. uniform crossover, can perform stably in the course of evolution.

Corollary 3.11. *Let e_{u} be the expected number of distinct genes in the offspring reproduced by n-parent U-Scan $\mathcal{X}_{\mathrm{u}}$. Subject to uniform population $U_{(k)}$, the expectation e_{u} is equal to the order k, regardless of the adopted number of parents n. Precisely, the expectation is*

$$e_{\mathrm{u}}(n, U_{(k)}) = k \quad \text{for } n \in \mathbb{N}_{>1}$$

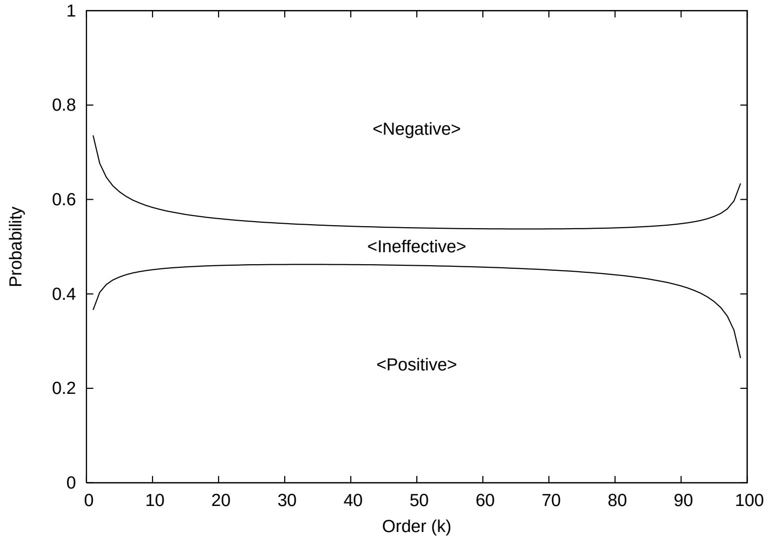


Fig. 3.3. Performance of U-Scan $\mathcal{F}(\mathcal{X}_{\mathrm{u}}, 2, U_{(k)})$

Proof. Let $\mathbf{c}'$ be the offspring reproduce by $\mathcal{X}_{\mathrm{u}}$, $D(\mathbf{c}')$ be the set of distinct genes in $\mathbf{c}'$, and $|D(\mathbf{c}')|$ be the cardinality of $D(\mathbf{c}')$. From (3.3) and (3.4), the expectation $e_{\mathrm{u}}(n, U_{(k)})$ can be derived:

$$\begin{aligned} e_{\mathrm{u}}(n, U_{(k)}) = \mathrm{E}\left[\left|D(\mathbf{c}')\right|\right] &= \sum_{\delta=0}^{l} \delta \cdot \Pr\left\{\left|D(\mathbf{c}')\right| = \delta\right\} \\ &= \sum_{\delta=0}^{l} \delta \cdot B\left(\delta; l, p_{\mathrm{u}}\right) = l \cdot p_{\mathrm{u}} = k. \end{aligned}$$

□

Corollary 3.11 shows that U-Scan is expected to yield the same number of distinct genes as the parents on average; that is to say, there is no tendency for U-Scan toward better or worse solutions with respect to the number of distinct genes.

3.3.2 Analysis of OB-Scan

Instead of random, OB-Scan determines an offspring gene depending on the occurrence of parental genes at that locus. Specifically, OB-Scan assigns the majority of parental genes to the offspring. In this section we will examine the performance of OB-Scan under the assumption of k-order uniform population.

Lemma 3.12. *Given n parents selected from $U_{(k)}$, the probability p_{ob} for OB-Scan $\mathcal{X}_{\text{ob}}$ to assign the distinct gene to an offspring locus is*

$$p_{\text{ob}} = \begin{cases} \sum_{x=\frac{n}{2}+1}^{n} B\left(x;n,p\right) + \frac{1}{2} B\left(\frac{n}{2};n,p\right) & \text{if } n \text{ is even,} \\ \sum_{x=\lceil \frac{n}{2} \rceil}^{n} B\left(x;n,p\right) & \text{otherwise.} \end{cases} \tag{3.8}$$

Proof. Assume x of n parents hold the distinct gene at locus $i \in \{1,\dots,l\}$. Since OB-Scan picks the majority of parental genes as the offspring gene, the probability for OB-Scan to assign the distinct gene to the offspring c'_i is

$$\Pr\left\{c'_i \text{ is a distinct gene} \mid x\right\} = \begin{cases} 1 & \text{if } x > \frac{n}{2}, \\ 0.5 & \text{if } x = \frac{n}{2}, \\ 0 & \text{if } x < \frac{n}{2}. \end{cases}$$

From Lemma 3.7 and Lemma 3.8 we know the p.m.f. of the variable x is $B\left(x;n,p\right)$ with $p = \frac{k}{l}$. The probability p_{ob} for $n \in 2\mathbb{N}$ can then be derived:

$$\begin{aligned} p_{\text{ob}} &= \sum_{x=0}^{n} \Pr\left\{c'_i \text{ is a distinct gene} \mid x\right\} \cdot \Pr(x) \\ &= \sum_{x=\frac{n}{2}+1}^{n} 1 \cdot B\left(x;n,p\right) + \frac{1}{2} \cdot B\left(\frac{n}{2};n,p\right) + \sum_{x=0}^{\frac{n}{2}-1} 0 \cdot B\left(x;n,p\right) \\ &= \sum_{x=\frac{n}{2}+1}^{n} B\left(x;n,p\right) + \frac{1}{2} B\left(\frac{n}{2};n,p\right). \end{aligned}$$

The probability p_{ob} for $n \in 2\mathbb{N}+1$ can be derived analogously. □

Theorem 3.13 (Performance of OB-Scan). *Given n parents selected from $U_{(k)}$, for the performance of OB-Scan $\mathcal{X}_{\text{ob}}$ we have*

$$\mathcal{F}(\mathcal{X}_{\text{ob}}, n, U_{(k)}) = \left(\sum_{i=0}^{k-1} B(i;l,p_{\text{ob}}), B(k;l,p_{\text{ob}}), \sum_{i=k+1}^{l} B(i;l,p_{\text{ob}}) \right),$$

where p_{ob} is the probability defined in Lemma 3.12.

Proof. Let $\mathbf{c}'$ be the offspring reproduced by $\mathcal{X}_{\text{ob}}$ and $|D(\mathbf{c}')|$ be the number of distinct genes in $\mathbf{c}'$. From Lemma 3.12, we have the probability p_{ob} that OB-Scan assigns the distinct gene to an offspring locus. In addition, scanning crossover processes each gene of an offspring independently. Processing l genes in this way, OB-Scan has the probability that δ distinct genes occur in the offspring $\mathbf{c}'$:

$$\Pr\left\{\left|D(\mathbf{c}')\right| = \delta\right\} = \binom{l}{\delta} \left(p_{\text{ob}}\right)^{\delta} \left(1 - p_{\text{ob}}\right)^{l-\delta} = B\left(\delta; l, p_{\text{ob}}\right). \tag{3.9}$$

This gives the probability p^+, $p^=$, and p^- of OB-Scan $\mathcal{X}_{\text{ob}}$, respectively:

$$\begin{aligned}
p^+(\mathcal{X}_{\text{ob}}, n, U_{(k)}) &= \Pr\left\{\left|D(\mathbf{c}')\right| < k\right\} = \sum_{\delta=0}^{k-1} B\left(\delta; l, p_{\text{ob}}\right) \\
p^=(\mathcal{X}_{\text{ob}}, n, U_{(k)}) &= \Pr\left\{\left|D(\mathbf{c}')\right| = k\right\} = B\left(k; l, p_{\text{ob}}\right) \\
p^-(\mathcal{X}_{\text{ob}}, n, U_{(k)}) &= \Pr\left\{\left|D(\mathbf{c}')\right| > k\right\} = \sum_{\delta=k+1}^{l} B\left(\delta; l, p_{\text{ob}}\right)
\end{aligned}$$

Hence the performance of OB-Scan is given by

$$\mathcal{F}(\mathcal{X}_{\text{ob}}, n, U_{(k)}) = (p^+, p^=, p^-) = \left(\sum_{i=0}^{k-1} B(i; l, p_{\text{ob}}), B(k; l, p_{\text{ob}}), \sum_{i=k+1}^{l} B(i; l, p_{\text{ob}})\right).$$

□

Corollary 3.14 (Pairwise Equivalence). *Subject to uniform population, the performance of* $(2a)$*-parent OB-Scan is equivalent to that of* $(2a-1)$*-parent OB-Scan with* $a \in \mathbb{N}_{>1}$*:*

$$\mathcal{F}\left(\mathcal{X}_{\text{ob}}, 2a, U_{(k)}\right) = \mathcal{F}\left(\mathcal{X}_{\text{ob}}, 2a-1, U_{(k)}\right).$$

Proof. Theorem 3.13 shows the performance of OB-Scan depends upon the probability p_{ob}. According to (3.8), the probabilities p_{ob} for $n = 2a$ and $n = 2a-1$ are respectively

$$\begin{aligned}
p_{\text{ob.odd}} &= 1 - \sum_{x=0}^{a-1} B\left(x; 2a-1, p\right) \qquad (3.10) \\
p_{\text{ob.even}} &= 1 - \sum_{x=0}^{a-1} B\left(x; 2a, p\right) - \frac{1}{2} B\left(a; 2a, p\right).
\end{aligned}$$

Since

$$B\left(x; n, p\right) = p \cdot B\left(x-1; n-1, p\right) + (1-p) \cdot B\left(x; n-1, p\right),$$

we have

$$\begin{aligned}
p_{\text{ob.even}} &= 1 - \sum_{x=0}^{a-1} \left[p \cdot B\left(x-1; 2a-1, p\right) + (1-p) \cdot B\left(x; 2a-1, p\right)\right] - \frac{1}{2} B\left(a; 2a, p\right) \\
&= 1 - \sum_{x=0}^{a-1} B\left(x; 2a-1, p\right) \\
&\quad - p \sum_{x=0}^{a-1} \left[B\left(x-1; 2a-1, p\right) - B\left(x; 2a-1, p\right)\right] - \frac{1}{2} B\left(a; 2a, p\right).
\end{aligned}$$

Using (3.10), the previous equation can be rewritten as

$$\begin{aligned}
p_{\text{ob.even}} &= p_{\text{ob.odd}} + p \cdot B\left(a-1; 2a-1, p\right) - \frac{1}{2} B\left(a; 2a, p\right) \\
&= p_{\text{ob.odd}} + p \frac{(2a-1)!}{(a-1)!a!} p^{a-1} \left(1-p\right)^{a} - \frac{1}{2} \frac{(2a)!}{a!a!} p^{a} \left(1-p\right)^{a} \\
&= p_{\text{ob.odd}} + \left[\frac{a}{2a} \frac{(2a)!}{a!a!} - \frac{1}{2} \frac{(2a)!}{a!a!} \right] p^{a} \left(1-p\right)^{a} \\
&= p_{\text{ob.odd}}.
\end{aligned}$$

This equivalence of $p_{\text{ob.even}}$ and $p_{\text{ob.odd}}$ gives identical probabilities p^{+}, $p^{=}$, and p^{-} for $(2a)$-parent OB-Scan and $(2a-1)$-parent OB-Scan. According to Theorem 3.13, we complete the proof of $\mathcal{F}(\mathcal{X}_{\text{ob}}, 2a, U_{(k)}) = \mathcal{F}(\mathcal{X}_{\text{ob}}, 2a-1, U_{(k)})$ for $a \in \mathbb{N}_{>1}$. □

Corollary 3.15. *Given n parents selected from $U_{(k)}$, the expected number e_{ob} of distinct genes in the offspring reproduced by OB-Scan is*

$$e_{\text{ob}}(n, U_{(k)}) = l \cdot p_{\text{ob}}. \tag{3.11}$$

Proof. Let $\mathbf{c}'$ be the offspring reproduce by $\mathcal{X}_{\text{ob}}$, $D(\mathbf{c}')$ be the set of distinct genes in $\mathbf{c}'$, and $|D(\mathbf{c}')|$ be the cardinality of $D(\mathbf{c}')$. From (3.3) and (3.9) we have the expectation

$$\begin{aligned}
e_{\text{ob}}(n, U_{(k)}) = \mathrm{E}\left[\left|D(\mathbf{c}')\right|\right] &= \sum_{\delta=0}^{l} \delta \cdot \Pr\left\{\left|D(\mathbf{c}')\right| = \delta\right\} \\
&= \sum_{\delta=0}^{l} \delta \cdot B\left(\delta; l, p_{\text{ob}}\right) \\
&= l \cdot p_{\text{ob}}.
\end{aligned}$$

□

Different from U-Scan, the performance of OB-Scan is dependent on the number of parents n as p_{ob} is dependent on n. Figure 3.4 respectively plots the performance p^{+}, $p^{=}$, and p^{-} of OB-Scan according to Theorem 3.13 for $l = 100$. Owing to the pairwise equivalence, here we only show the performance of OB-Scan with even numbers of parents. As the figures show, OB-Scan with more parents accounts for a higher probability (even approximates to 1.0) to reproduce k-positive offspring, as the order k is smaller than a half of chromosome length ($k < \frac{l}{2} = 50$). On the other hand, it accounts for a higher probability to yield k-negative offspring when the order is larger than a half of chromosome length ($k > 50$). Additionally, the area of *transition phase* around $k = \frac{l}{2}$ becomes narrower as the number of parents increases. That is to say, in terms of the number of distinct genes, OB-Scan tends to reproduce better offspring as $k < \frac{l}{2}$ while it tends to reproduce worse offspring as $k > \frac{l}{2}$. Raising parents, furthermore, increases the influence of order k upon the tendency of OB-Scan to reproduce better or worse offspring.

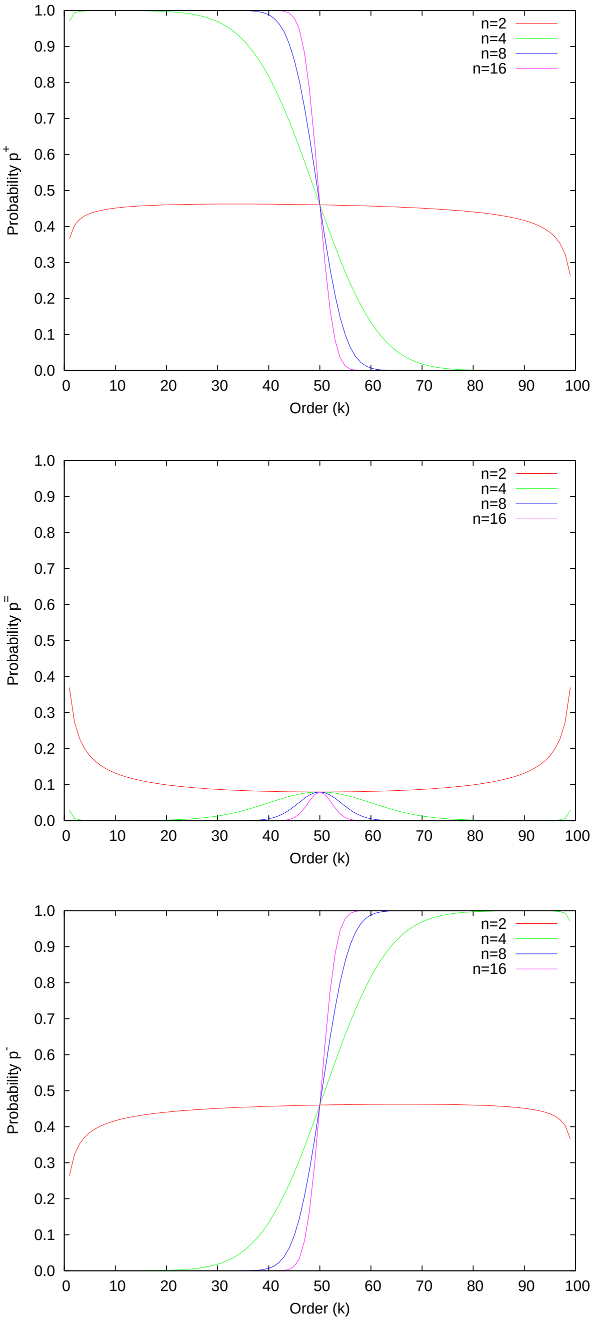


Fig. 3.4. Performance of OB-Scan $\mathcal{F}(\mathcal{X}_{\mathrm{ob}}, n, U_{(k)})$ in terms of order k for $l = 100$

Figure 3.5 shows the expected number of distinct genes $e_{\mathrm{ob}}(n, U_{(k)})$ using Corollary 3.15 for $l = 100$. The profile of 2-parent OB-Scan, i.e. uniform crossover, demonstrates that offspring will averagely inherit the same amount of distinct genes as their parents. Raising parents in OB-Scan, however, causes two opposing effects. As the order k is smaller than $\frac{l}{2}$, OB-Scan is expected to yield solutions with fewer distinct genes, namely better solutions. On the contrary, if the order k is larger than $\frac{l}{2}$, OB-Scan will reproduce worse offspring. Nonetheless, the difference in expectation $e_{\mathrm{ob}}(n, U_{(k)})$ caused by increasing parents gradually diminishes as n is increased. In Fig. 3.4, we can find a similar trend in the probabilities p^+, $p^=$, and p^-.

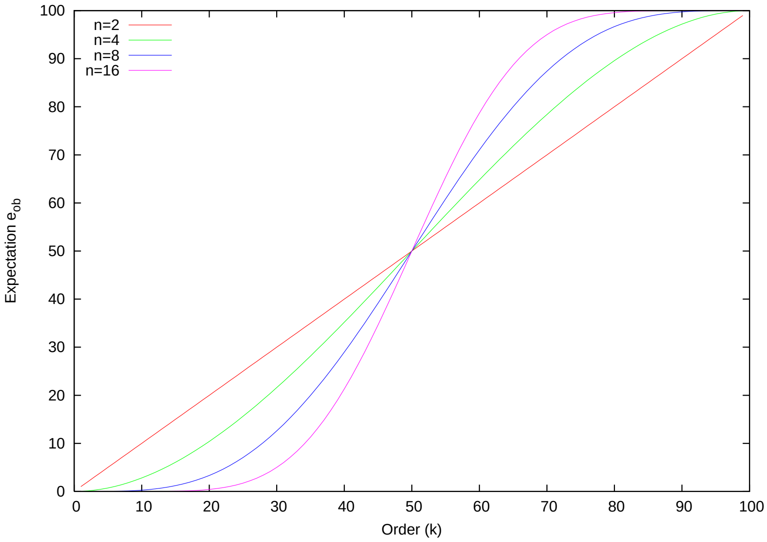


Fig. 3.5. Expected number $e_{\mathrm{ob}}(n, U_{(k)})$ of distinct genes in the offspring for $l = 100$

At first glance, raising parents in OB-Scan seems advantageous to performance because of its expected reduction of distinct genes. However, we should carefully consider the case that the order k is larger than a half of chromosome length. In this case, OB-Scan is less likely to reproduce better offspring. In addition, raising parents in OB-Scan reinforces the improvement or the deterioration in 2-parent OB-Scan. As above-stated, this reinforcement will tend to be steady as n is large. The foregoing phenomenons can be summed up in the following conjecture.

Conjecture 3.16. Raising parents in OB-Scan intensifies exploitation of the search.

The conventional wisdom tells that intensified exploitation accelerates the search; on the other hand, it may cause premature convergence. Our conjecture therefore anticipates that raising parents in OB-Scan will receive an accelerated but likely premature convergence. Furthermore, it reveals the necessity of balancing exploration and exploitation when using OB-Scan with more than two parents.

3.4 Experimental Validation

In this section, we conduct a series of experiments to validate our theoretical analysis of U-Scan and OB-Scan. Four common test functions are used as benchmarks: the extended De Jong's F2 (F2e), the Rastrigin (RAS), the Schwefel (SCH), and the Griewangk (GRI) functions. Table 3.1 lists the properties and the parameters of these functions[3]. The setting of MPGA employed in our experiments is listed in Table 3.2. Note that two to twenty parents are adopted in both crossovers to identify the influence of the number of parents. In addition, the survival strategy is to delete the worst chromosome with 'no duplicates' policy. Each experiment includes 100 independent runs.

Table 3.1. Test functions

Function	N	Bits of x_i	l
$f_{\text{F2e}} = \sum_{i=1}^{N-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right], x_i \in [-2.048, 2.047]$	10	12	120
$f_{\text{RAS}} = 10N + \sum_{i=1}^{N} \left[x_i^2 - 10\cos(2\pi x_i)\right], x_i \in [-5.12, 5.11]$	10	10	100
$f_{\text{SCH}} = 418.98291N + \sum_{i=1}^{N} -x \sin\left(\sqrt{\lvert x_i \rvert}\right), x_i \in [-512, 511]$	10	10	100
$f_{\text{GRI}} = 1 + \sum_{i=1}^{N} \frac{x_i^2}{4000} - \prod_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right), x_i \in [-512, 511]$	10	10	100

Table 3.2. The setting of MPGAs in experiments

GA type	Steady-state GA
Representation	Bit string
Population size	100
Selection	Linear ranking selection with bias 1.25
Crossover	U-Scan / OB-Scan with $n = 2, \ldots, 20$
Crossover rate	1.0
Mutation	Bit-flip mutation
Mutation rate	$\frac{1}{l}$

Figure 3.6 compares the mean of the best fitness obtained from MPGAs using U-Scan and MPGAs using OB-Scan with different numbers of parents on the four test functions. These experimental results show a high level of consistence with our theoretical arguments: First, the profiles of U-Scan in Fig. 3.6 remain constant for different numbers of parents, which corresponds to Theorem 3.10: the number of parents has no effect on the performance of U-Scan. Second, as Theorem 3.14 indicated, the profiles of OB-Scan in Fig. 3.6 show an analogy between odd numbers and even numbers of

[3] A more detailed description of these functions is given in Appendix A.

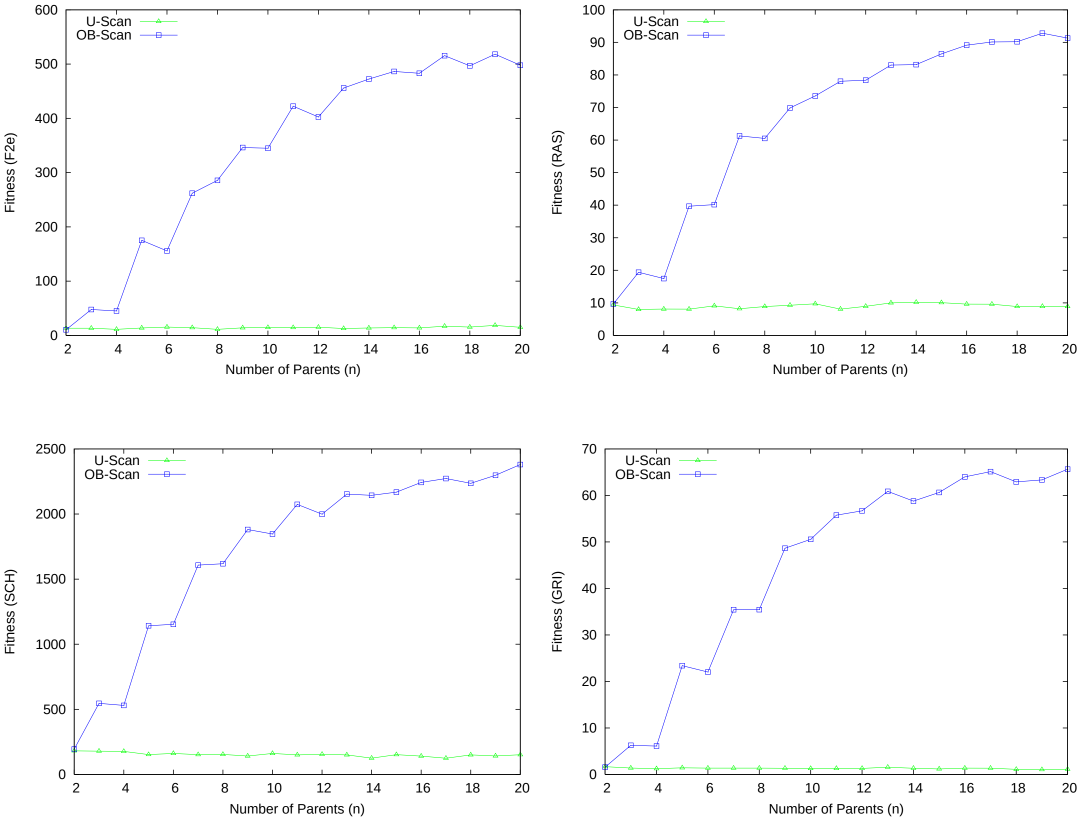


Fig. 3.6. The mean of the best fitness for U-Scan and OB-Scan with different numbers of parents over 100 runs

parents in solution quality. Moreover, the solution quality of OB-Scan deteriorates as the number of parents increases.

We further perform a two-tailed t-test to examine whether there exists statistically significant difference in solution quality for MPGAs using more parents. Table 3.3 shows the results of a t-test on the best solutions of MPGAs using 2 and n parents for $n = 3, \ldots, 20$. With significance level $\alpha = 0.01$, the P-values in this table demonstrate that, in general, the difference in solution quality between 2-parent U-Scan and n-parent U-Scan is not significant. This table also validates that raising parents in OB-Scan causes a highly significant deterioration in solution quality, compared with 2-parent OB-Scan. Additionally, Table 3.4 compares the best solution of MPGAs using $n-1$ and n parents for $n = 3, \ldots, 20$. This table shows no significant difference in solution quality between $(n-1)$-parent U-Scan and n-parent U-Scan, yet the P-values of OB-Scan exhibit periodic significance as $n < 13$. This periodic equivalence supports the *pairwise equivalence* claimed in Theorem 3.14. As aforementioned, the influence of the increase of parents will tend to be steady; thus the difference for $n \geq 13$ is less likely to

Table 3.3. P-values of t-test ($\alpha = 0.01$) on the solution quality between 2 and n parents (only significant values in U-Scan are marked by boldface)

	U-Scan				OB-Scan			
2 vs. n	F2e	RAS	SCH	GRI	F2e	RAS	SCH	GRI
3	.9918	**.0033**	.8875	.2492	5.89e-31	7.76e-31	2.31e-36	1.08e-28
4	.3654	.0311	.8132	.0531	1.35e-12	2.28e-20	3.27e-30	4.13e-22
5	.8711	**.0073**	.1075	.2990	1.49e-39	1.73e-58	6.94e-68	2.09e-44
6	.4387	.7155	.2434	.1896	7.30e-40	1.84e-58	9.13e-67	1.79e-40
7	.7143	.0224	.0801	.2107	7.18e-44	2.35e-70	2.11e-85	1.28e-50
8	.3454	.4994	.1156	.1944	6.40e-45	1.41e-73	1.06e-89	1.49e-51
9	.8068	.9060	.0215	.1528	3.13e-45	9.37e-75	9.73e-91	7.39e-57
10	.6269	.6848	.2469	.0866	4.27e-42	4.23e-82	1.65e-90	7.28e-62
11	.6950	**.0052**	.0825	.1203	6.22e-40	6.52e-83	1.51e-102	5.45e-61
12	.4953	.5202	.0961	.1009	1.11e-43	3.76e-82	7.95e-104	5.92e-59
13	.8186	.5009	.0776	.6734	1.53e-43	1.54e-85	6.56e-103	3.24e-59
14	.8539	.3605	**.0009**	.1765	2.27e-39	1.58e-89	3.71e-97	1.65e-65
15	.7102	.4948	.0971	.0276	3.88e-47	2.14e-89	5.11e-93	1.11e-62
16	.8192	.7844	.0223	.1914	1.25e-38	4.18e-90	5.01e-106	6.45e-65
17	.1845	.7419	**.0007**	.1577	9.64e-46	1.96e-94	1.25e-106	2.07e-61
18	.4465	.3070	.0772	.0122	4.17e-46	7.24e-98	3.48e-102	2.15e-59
19	.0763	.3645	.0232	**.0062**	1.95e-45	3.30e-92	9.20e-117	1.17e-58
20	.5537	.5181	.0689	.0173	4.85e-43	1.26e-91	6.00e-111	7.43e-59

be significant. The statistical results shown in Tables 3.3 and 3.4 therefore reconfirm the effectiveness of our analytical claims in solution quality.

Next, we examine the convergence of MPGAs using U-Scan and OB-Scan individually. Figure 3.7 plots the convergence for U-Scan and OB-Scan with different numbers of parents. Here only the result in SCH is presented owing to the similarity of profiles in the four test functions. Obviously, the overlapping convergence of U-Scan in Fig. 3.7 restates that the number of parents is ineffective in the performance of U-Scan. On the other hand, the convergence of OB-Scan strongly depends upon the adopted number of parents: The profiles of OB-Scan with 3 and 4 parents are very similar, and so are those with 5 and 6 parents, and with 7 and 8 parents — these similarities validate Theorem 3.14. Furthermore, Fig. 3.7 demonstrates that increasing parents accelerates convergence in the beginning but thereafter gets trapped in premature convergence, which is a classic characteristic of overly-intensified exploitation. This outcome confirms our conjecture: raising parents in OB-Scan will intensify exploitation of the search. Consequently, this intensified exploitation causes premature convergence of OB-Scan in the test functions.

3.5 Summary

This chapter gave a formal analysis of two scanning crossovers, which are multi-parent generalizations of uniform crossover. First we proposed a simplified population model, the *uniform population* model, to provide a systematic population environment for

Table 3.4. P-values of t-test ($\alpha = 0.01$) on the solution quality between n and $n-1$ parents (all significant values are marked by boldface)

	U-Scan				OB-Scan			
n-1 vs. n	F2e	RAS	SCH	GRI	F2e	RAS	SCH	GRI
2 - 3	.9918	**.0033**	.8875	.2492	**.0000**	**.0000**	**.0000**	**.0000**
3 - 4	.3588	.8314	.9281	.5137	.6471	.0220	.5543	.7480
4 - 5	.2629	.9869	.1650	.3210	**.0000**	**.0000**	**.0000**	**.0000**
5 - 6	.5231	.1754	.6135	.7390	.0579	.7198	.7699	.2906
6 - 7	.6823	.2410	.5643	.9706	**.0000**	**.0000**	**.0000**	**.0000**
7 - 8	.1845	.3332	.9153	.9843	.1188	.6444	.7905	.9926
8 - 9	.2048	.6673	.4769	.9139	**.0006**	**.0000**	**.0000**	**.0000**
9 - 10	.7933	.6675	.2493	.7771	.9502	.0392	.4272	.3143
10 - 11	.9312	.0561	.5450	.9255	**.0013**	.0106	**.0000**	**.0078**
11 - 12	.7706	.1473	.7997	.9679	.4223	.8611	.0593	.6612
12 - 13	.3460	.3016	.7859	.2176	.0303	.0108	**.0002**	.0648
13 - 14	.6636	.8808	.1336	.3382	.5572	.9303	.8287	.3299
14 - 15	.8378	.9222	.1113	.4425	.6151	.0581	.5913	.3638
15 - 16	.8811	.7487	.5378	.3272	.9056	.1307	.0944	.1188
16 - 17	.2685	.9771	.3154	.9057	.2761	.5796	.4744	.6175
17 - 18	.5843	.2979	.1153	.2135	.4915	.9630	.3919	.3556
18 - 19	.3139	.9381	.6242	.7165	.4342	.1237	.1282	.8630
19 - 20	.2434	.9744	.5750	.5782	.4763	.3998	.0375	.3567

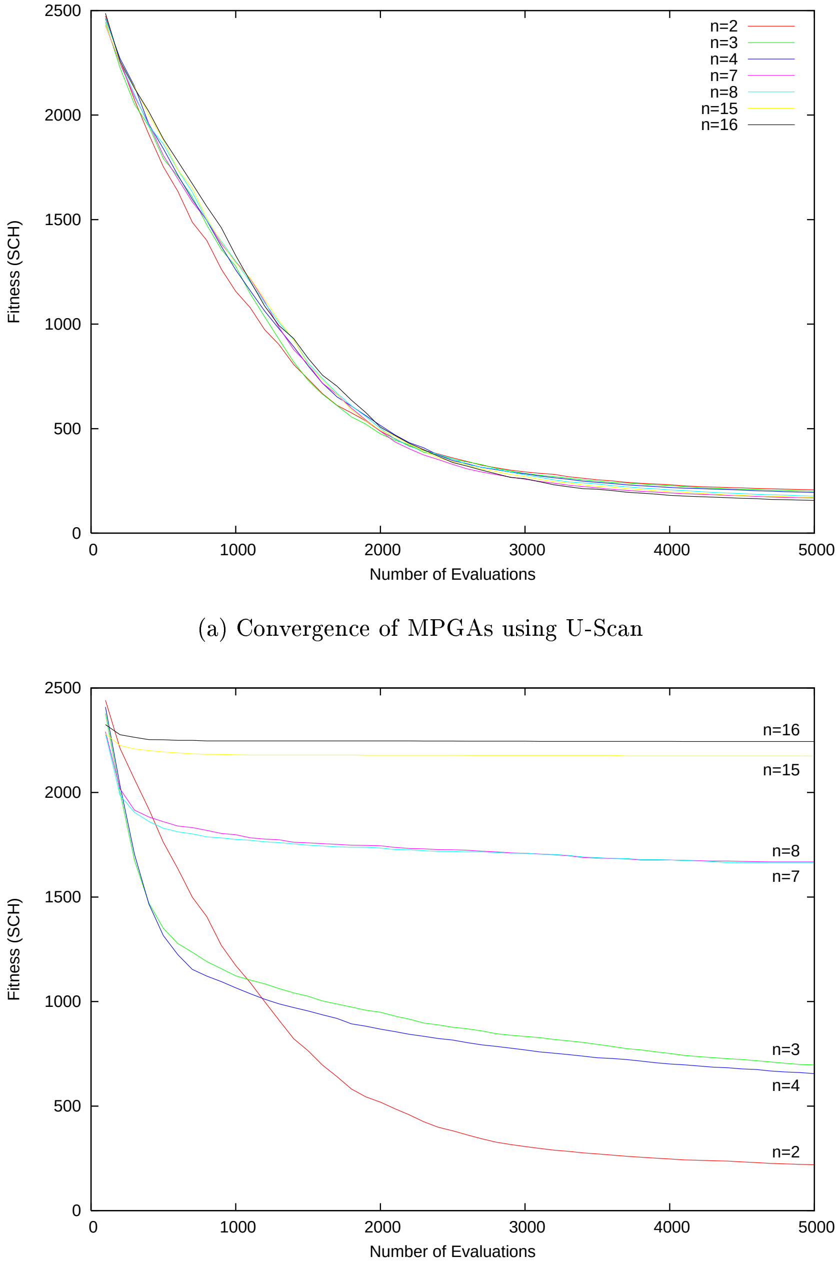
analysis. A criterion based on uniform population was further presented to evaluate the performance of crossover. Accordingly we analyzed U-Scan and OB-Scan to identify the influence of applying more than two parents on their performance.

Several interesting results emerge out of our analysis. Concerning the question "*When and why will MPGAs perform better?*", first, we proved that the number of parents exercises no influence upon the performance of U-Scan. That is to say, U-Scan with more than two parents will perform the same as its 2-parent version, i.e. uniform crossover. Second, contrary to U-Scan, the number of parents plays an important role in the performance of OB-Scan. Under the assumption of uniform population, OB-Scan with an even number n of parents is proved to perform equally to OB-Scan with $n-1$ (an odd number) parents. In addition, the analysis reveals that using more than two parents in OB-Scan will intensively lead to better solutions, as the order of uniform population is smaller than a half of chromosome length; on the other hand, it will yield worse offspring as the order is larger than a half of chromosome length. Furthermore, the analysis shows that increasing the number of parents will further reinforce the improvement (or deterioration). According to the analytical results of OB-Scan, a conjecture is made that raising parents in OB-Scan will intensify exploitation of the search.

The analytical claims are further verified by a series of experiments. We applied MPGAs with U-Scan and OB-Scan individually to solve four common test functions. Experimental results show a high level of consistence with our analytical arguments: First, raising parents in U-Scan, as expected, makes no significant difference with 2-

parent U-Scan in both solution quality and convergence. Second, OB-Scan shows pairwise equivalence in solution quality as well as convergence. Moreover, experimental results reveal that increasing parents in OB-Scan causes an accelerated but premature convergence — a common consequence of overly-intensified exploitation. In a word, these experimental results validate our analytical claims and conjecture.

To some extent the proposed uniform population is too simple to represent the real population; nonetheless, it provides a tractable population environment and a convenient way to analyze crossover theoretically. Furthermore, the analytical results based on uniform population are capable of identifying the influence of applying more parents on the performance of multi-parent crossovers. However, the present analysis concentrates on one generation and only gives the trend for a MPGA toward better, equal, or worse performance. Beyond that, the following chapter will extend the notion of this chapter to an advanced analysis on the correlation between the number of parents and the convergence of MPGAs, in consideration of the whole evolution and the regular population.



(a) Convergence of MPGAs using U-Scan

(b) Convergence of MPGAs using OB-Scan

Fig. 3.7. The convergence of MPGAs using U-Scan and OB-Scan with different numbers of parents in the Schwefel function

4

Analysis Using Markov Chain Theory

This chapter models MPGAs by Markov chain theory to deal with the following two issues:

- When will MPGAs outperform GAs?
- Why do MPGAs perform better?

First, we look into the variation of gene frequency affected by selection, multi-parent crossover, and mutation individually. Following the framework of MPGAs, the gene frequency at each stage (selection, crossover, and mutation) can be derived. Next, we formulate the variation of gene frequency over generations by Markov chain theory. Here the survivor strategy is considered. As a result, the separate as well as the integral effects of *population size*, *selection intensity*, *the number of parents*, *mutation rate*, and *generation gap* on the gene frequency over generations are modeled. The evolution in MPGAs, as Fig. 1.1 illustrates, is comprehensively analyzed.

The proposed Markov model can afford to resolve the above two issues in MPGAs. First, we examine the genetic drift, the principal measure of how fast population diversity diminishes. To this end, no mutation, i.e. zero mutation rate, is applied to the model. The consequent Markov model gives the definite number of generations when population diversity will drain away. That is to say, it explicitly indicates the speed of genetic drift in MPGAs. Second, we use the Markov model to investigate the convergence of MPGAs in the OneMax problem; furthermore, we extend this model to deal with the Generalized OneMax problem. The proposed Markov model not only indicates the correlation between the number of parents and the expected fitness, but also identifies the impact of other MPGA operators. Consequently, this model can point out whether MPGAs can outperform GAs with respect to a particular parameter setting, which helps us to explore the situations and the causes of MPGAs' superiority over GAs.

Part of the content presented in this chapter was published in [103, 104].

4.1 Variation of Gene Frequency

Gene frequency represents the proportion of a particular allele in the population and is widely used as a quantitative measure of genetic variation in population genetics [53, 75]. It also suffices to clue us in on the course of evolution in GAs. In this section we analyze the variation of gene frequency caused by U-Scan, OB-Scan, and mutation respectively. Based on the gene frequency, the next section will build the model with Markov chain theory.

Definition 4.1 (Gene Frequency). *The* gene frequency $p_k(\alpha, t)$ *is defined as the proportion of allele α at locus k in the population at time t. Let $C = \{\mathbf{c}_1, \ldots, \mathbf{c}_m\}$ be the population at time t and let $C_k(\alpha) = \{\mathbf{c} \in C \mid c_k = \alpha\}$ be the subset in which chromosomes possess allele α at locus k. The gene frequency is defined by*

$$p_k(\alpha, t) \overset{\text{def}}{=} \frac{|C_k(\alpha)|}{|C|},$$

where $|C|$and $|C_k(\alpha)|$ represent the cardinality of set C and $C_k(\alpha)$.

In this thesis, chromosomes in GAs are represented as binary strings. Thus there exist two gene frequencies $p_k(1, t)$ and $p_k(0, t)$ with $p_k(0, t) = 1 - p_k(1, t)$ for locus k at time t. For simplicity, we refer to the gene frequency $p_k(1, t)$ as $p_k(t)$ and refer to $p_k(0, t)$ as $(1 - p_k(t))$. Incidentally, the symbol $p_k(t)$ is referred to as p_k whenever the indication of time t is irrelevant.

Remark 4.2. In GAs, we have

$$p_k \overset{\text{def}}{=} \mathrm{E}[c_k] = \frac{1}{m} \sum_{\mathbf{c} \in C} c_k, \tag{4.1}$$

$$\sigma_k^2 \overset{\text{def}}{=} \mathrm{Var}(c_k) = p_k(1 - p_k), \tag{4.2}$$

where $\mathrm{E}[\cdot]$ denotes the expectation and $\mathrm{Var}(\cdot)$ denotes the variance.

Definition 4.3 (Variation of Gene Frequency in GAs). *Let $p_k^s(t)$, $p_k^x(t)$, $p_k^m(t)$ be the gene frequencies after performing selection, crossover, and mutation at generation t. The process of GAs with respect to gene frequency can be expressed as*

$$p_k(t) \xrightarrow{\text{selection}} p_k^s(t) \xrightarrow{\text{crossover}} p_k^x(t) \xrightarrow{\text{mutation}} p_k^m(t) \xrightarrow{\text{survivor}} p_k(t+1). \tag{4.3}$$

The variation of gene frequency caused by U-Scan, OB-Scan, and bit-flip mutation will be analyzed in the subsequent sections. The effect of the survivor strategy on the gene frequency will be examined in Section 4.3 and Section 4.4.3. The selection, however, is associated with the fitness of chromosomes while the fitness depends upon the given problem. This implies that the influence of selection on the gene frequency is problem-dependent. We will investigate this influence in the OneMax problem in Section 4.4 and the influence in the Generalized OneMax problem in Section 4.5. Before that, we simply assume the selection is a random selection; that is, the gene frequency $p_k^s(t) = p_k(t)$.

In the analysis we will make use of two well-known probabilistic distributions [79]. First, the probability mass function (p.m.f) $B(x; n, p)$ of *binomial distribution* $B(n, p)$ is

$$B(x; n, p) = \binom{n}{x} p^x (1-p)^{n-x},$$

where $n, x \in \mathbb{Z}_*$ and and $0 \leq p \leq 1$. Second, the p.m.f. $H(x; n, r, m)$ of *hypergeometric distribution* $H(n, r, m)$ for $x, n, r, m \in \mathbb{Z}_*$ is

$$H(x; n, r, m) = \frac{\binom{r}{x}\binom{m-r}{n-x}}{\binom{m}{n}}.$$

4.1.1 Variation Caused by U-Scan

Section 1.2 has introduced how U-Scan and OB-Scan operate. Here we analyze the gene frequency affected by U-Scan. In the next section we will further investigate the gene frequency affected by OB-Scan. Since both U-Scan and OB-Scan are multi-parent generalizations of uniform crossover, the analyses and results presented here are applicable to uniform crossover as well.

Lemma 4.4. *Suppose we have the gene frequency p_k^s of the selected parents. For the gene frequency, denoted by p_k^{u}, of the offspring reproduced by n-parent U-Scan $\mathcal{X}_{\mathrm{u}}$ with $n \in \mathbb{N}_{>1}$, we have*

$$p_k^{\mathrm{u}} = p_k^s.$$

Proof. Let X be the number of parents possessing the allele 1 at locus k among n selected parents. Since the process of selection is independent, i.e. the i^{th} selection exercising no influence on the j^{th} selection, the process is a Bernoulli process. Performing this selection n times, the number X will hold a binomial distribution with p.m.f.

$$\Pr(X = x) = B(x; n, p_k^s) = \binom{n}{x} \left(p_k^s\right)^x \left(1 - p_k^s\right)^{n-x}.$$

Let $\mathfrak{D}_1$ denote the event that U-Scan assigns the allele 1 to the offspring locus k. Since U-Scan chooses the donor randomly, the number x out of n parents possessing the allele 1 yields

$$\Pr(\mathfrak{D}_1 \mid X = x) = \frac{x}{n}.$$

Hence the gene frequency

$$\begin{aligned} p_k^{\mathrm{u}} = \Pr(\mathfrak{D}_1) &= \sum_{x=0}^{n} \Pr(\mathfrak{D}_1 \mid X = x) \cdot \Pr(X = x) \\ &= \sum_{x=0}^{n} \left(\frac{x}{n}\right) B(x; n, p_k^s), \end{aligned}$$

Since the expectation of binomial distribution $\sum_x xB(x;n,p) = np$, the above equation becomes

$$p_k^{\mathrm{u}} = \frac{1}{n}(np_k^s) = p_k^s.$$

□

Corollary 4.5. *The number of parents in U-Scan has no influence upon the gene frequency.*

The above corollary can be easily proved by Lemma 4.4: U-Scan yields the same gene frequency p_k^{u} with the given p_k^s, no matter how many parents are adopted. That is to say, in terms of gene frequency, n-parent U-Scan performs identically with its 2-parent degeneration, viz uniform crossover. In Section 4.4, we will show that this ineffectiveness yields identical mean fitness at any time t and identical mean convergence time for different numbers of parents adopted in U-Scan.

4.1.2 Variation Caused by OB-Scan

Before conducting the analysis of OB-Scan, we introduce the incomplete beta function for simplifying the expression of equations.

Definition 4.6 (Incomplete Beta Function). *The* incomplete beta function *is defined as*

$$I_x(a,b) \overset{\mathrm{def}}{=} \frac{1}{\mathrm{Beta}(a,b)} \int_0^x t^{a-1}(1-t)^{b-1}dt,$$

where $a, b > 0$ and $\mathrm{Beta}(a,b)$ *is the* beta function.

The following properties hold for the incomplete beta function:

1. (26.5.24 [1]) For the binomial distribution $B(n,p)$,

$$\sum_{i=a}^{n} B(i;n,p) = I_p(a, n-a+1). \tag{4.4}$$

2. (26.5.16 [1])

$$I_x(a,b) = \frac{1}{a \cdot \mathrm{Beta}(a,b)} x^a(1-x)^b + I_x(a+1,b). \tag{4.5}$$

Now we embark on the analysis of OB-Scan's impact on the gene frequency.

Lemma 4.7. *Suppose we have the gene frequency p_k^s of the selected parents. For the gene frequency, denoted by p_k^{ob}, of the offspring reproduced by n-parent OB-Scan $\mathcal{X}_{\mathrm{ob}}$ with $n \in \mathbb{N}_{>1}$, we have*

$$p_k^{\mathrm{ob}} = I_{p_k^s}(a,a),$$

where I_p denotes the incomplete beta function and $a = \left\lceil \frac{n}{2} \right\rceil$.

Proof. Let X be the number of parents possessing the allele 1 at locus k among n selected parents. Since the process of selection is independent, it is a Bernoulli process. Performing this selection n times, the number X holds a binomial distribution with p.m.f.

$$\Pr(X = x) = B(x; n, p_k^s) = \binom{n}{x} (p_k^s)^x (1 - p_k^s)^{n-x} .$$

Let $\mathfrak{D}_1$ denote the event that OB-Scan assigns the allele 1 to the offspring locus k. According to Definition 1.5, OB-Scan yields

$$\Pr(\mathfrak{D}_1 \mid X = x) = \begin{cases} 1 & \text{if } x > n/2, \\ 0 & \text{if } x < n/2, \\ \frac{1}{2} & \text{if } x = n/2. \end{cases}$$

For OB-Scan with an odd number of parents $n = 2a-1$ for $a \in \mathbb{N}_{>1}$ (remark: $a = \lceil \frac{n}{2} \rceil$),

$$\begin{aligned} p_k^{\text{ob}} = \Pr(\mathfrak{D}_1) &= \sum_{x=0}^{n} \Pr(\mathfrak{D}_1 \mid X = x) \cdot \Pr(X = x) \\ &= \sum_{x=a}^{2a-1} B(x; 2a-1, p_k^s) \\ &= I_{p_k^s}(a, a) \qquad \text{(from (4.4))} \end{aligned}$$

Similarly, for OB-Scan with an even number of parents $n = 2a$ for $a \in \mathbb{N}$ (remark: $a = \lceil \frac{n}{2} \rceil$),

$$\begin{aligned} p_k^{\text{ob}} &= \sum_{x=0}^{n} \Pr(\mathfrak{D}_1 \mid X = x) \cdot \Pr(X = x) \\ &= \sum_{x=a+1}^{2a} B(x; 2a, p_k^s) + \frac{1}{2} B(a; 2a, p_k^s) \\ &= I_{p_k^s}(a+1, a) + \frac{1}{2}\binom{2a}{a} (p_k^s)^a (1 - p_k^s)^a \\ &= I_{p_k^s}(a, a) - \frac{1}{a\text{Beta}(a, a)} (p_k^s)^a (1 - p_k^s)^a \\ &\quad + \frac{1}{2}\binom{2a}{a} (p_k^s)^a (1 - p_k^s)^a \qquad \text{(from (4.5))} \\ &= I_{p_k^s}(a, a) + \left[-\frac{\Gamma(2a)}{a\Gamma(a)\Gamma(a)} + \frac{1}{2}\frac{(2a)!}{a!a!} \right] (p_k^s)^a (1 - p_k^s)^a \\ &= I_{p_k^s}(a, a) + \left[-\frac{a}{2a}\frac{(2a)!}{a!a!} + \frac{1}{2}\frac{(2a)!}{a!a!} \right] (p_k^s)^a (1 - p_k^s)^a \\ &= I_{p_k^s}(a, a). \end{aligned}$$

□

Comparing the gene frequencies associated with U-Scan and OB-Scan in Lemma 4.4 and Lemma 4.7, we obtain the correlation between p_k^{u} and p_k^{ob} as follows.

Corollary 4.8. *Let* $p_k^{\mathrm{ob}(n)}$ *and* $p_k^{\mathrm{u}(n)}$ *be the gene frequencies corresponding to* n*-parent OB-Scan and* n*-parent U-Scan, respectively. For any* $n \in \mathbb{N}_{>1}$*, we have*

$$p_k^{\mathrm{u}(n)} = p_k^{\mathrm{ob}(2)} = p_k^s.$$

This corollary indicates that U-Scan with *any* number of parents corresponds to OB-Scan with *2* parents in the gene frequency. Hence, U-Scan can be viewed as a special case of OB-Scan with $n = 2$. According to this correlation, in the subsequent analyses we will only discuss OB-Scan. Some more properties of OB-Scan in the gene frequency are presented below.

Corollary 4.9 (Pairwise Equivalence). *Let* $p_k^{\mathrm{ob}(n)}$ *be the gene frequency* p_k^{ob} *corresponding to* n*-parent OB-Scan. For* $n \in 2\mathbb{N}$ *and* $n \geq 4$*, we have*

$$p_k^{\mathrm{ob}(n)} = p_k^{\mathrm{ob}(n-1)}.$$

Proof. Trivial (since $\lceil \frac{n}{2} \rceil = \lceil \frac{n-1}{2} \rceil$ for $n \in 2\mathbb{N}$ in Lemma 4.7). □

Corollary 4.10. *Let* $p_k^{\mathrm{ob}(n)}$ *be the gene frequency* p_k^{ob} *corresponding to* n*-parent OB-Scan. For* $n \in \mathbb{N}_{>1}$*, it holds for the gene frequency* $p_k^{\mathrm{ob}(n+2)}$ *that*

$$p_k^{\mathrm{ob}(n+2)} \begin{cases} < p_k^{\mathrm{ob}(n)} & \textit{if } 0 < p_k^s < 0.5, \\ > p_k^{\mathrm{ob}(n)} & \textit{if } 0.5 < p_k^s < 1, \\ = p_k^{\mathrm{ob}(n)} & \textit{otherwise.} \end{cases}$$

Proof. Refer to Appendix D.1. □

Corollary 4.11. *For* $n \in \mathbb{N}_{>2}$*, it holds for the gene frequency* p_k^{ob} *that*

$$p_k^{\mathrm{ob}} \begin{cases} < p_k^s & \textit{if } 0 < p_k^s < 0.5, \\ > p_k^s & \textit{if } 0.5 < p_k^s < 1, \\ = p_k^s & \textit{otherwise.} \end{cases}$$

Proof. Refer to Appendix D.2. □

Figure 4.1 plots the differential of gene frequency $(p_k^{\mathrm{ob}} - p_k^s)$. As indicated in Corollary 4.11, the differential is divided by the gene frequency $p_k^s = 0.5$ into two parts: negative differential for $0 < p_k^s < 0.5$ and positive differential for $0.5 < p_k^s < 1$. In addition, raising parents in OB-Scan intensifies the tendency toward allele 0 or 1, depending on the given gene frequency p_k^s.

The intensification induced by OB-Scan can be beneficial if at all, one can select parents with a preference always for those owning the *correct* genes. Otherwise, OB-Scan with more than two parents may be harmful. This two-sided effect reveals the sensitivity of OB-Scan to the selection, and this sensitivity is further aggravated by the increase of parents. However, it is less likely to have a perfect selection for every locus, especially in the highly epistatic problems [55]. Therefore, a mechanism, e.g. mutation, that can remedy the harm caused by OB-Scan will be vital to the overall performance.

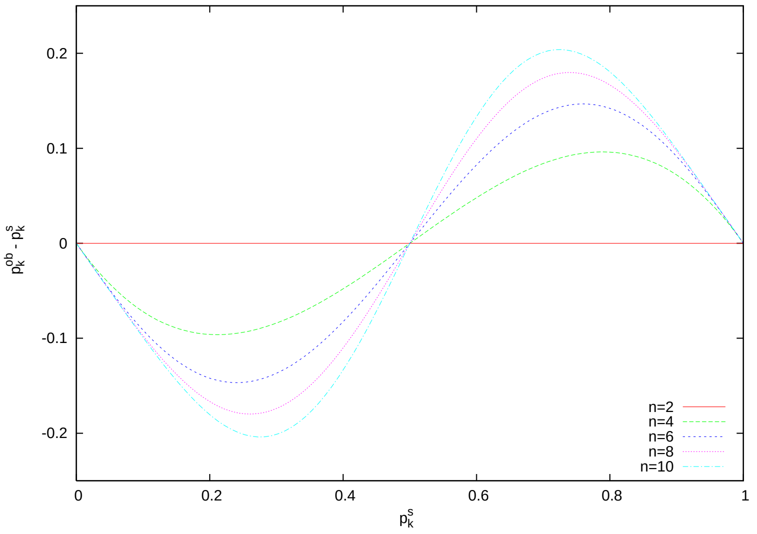


Fig. 4.1. The differential of gene frequency ($p_k^{\mathrm{ob}} - p_k^s$) induced by performing n-parent OB-Scan with p_k^s

4.1.3 Variation Caused by Mutation

In this chapter, the analysis of mutation focuses on the most popular mutation — *bit-flip mutation*. Bit-flip mutation randomly chooses a locus and then *flips* the gene at this locus, i.e. 0 → 1 and 1 → 0. A parameter γ_m, called *mutation rate*, is introduced to determine the probability for a gene to be mutated. The variation of gene frequency caused by bit-flip mutation is shown in the following lemma.

Lemma 4.12. *Suppose we have the gene frequency* p_k^x. *Given the mutation rate* γ_m, *for the gene frequency of the offspring mutated by bit-flip mutation, we have*

$$p_k^m = p_k^x + \gamma_m(1 - 2p_k^x).$$

Proof. Considering bit-flip mutation, the cases to yield the gene 1 at locus k are: $0 \rightarrow 1$ (mutated) and $1 \rightarrow 1$ (not mutated). Let c_k be the gene at locus k before mutation and c_k' be the gene after mutation. The gene frequency is

$$\begin{aligned} p_k^m &= \Pr\{c_k' = 1 \mid c_k = 0\} \Pr\{c_k = 0\} + \Pr\{c_k' = 1 \mid c_k = 1\} \Pr\{c_k = 1\} \\ &= \gamma_m(1 - p_k^x) + (1 - \gamma_m) p_k^x \\ &= p_k^x + \gamma_m(1 - 2p_k^x). \end{aligned}$$

□

Figure 4.2 plots the differential of gene frequency ($p_k^m - p_k^x$) induced by bit-flip mutation. Comparing Figs. 4.1 and 4.2, we learn that mutation exerts an influence contrary to OB-Scan in the gene frequency: For $0 < p_k^x < 0.5$, mutation has a positive differential, i.e. $p_k^m > p_k^x$. On the other hand, for $0.5 < p_k^x < 1$, mutation results in $p_k^m < p_k^x$. Simply speaking, mutation intends to pull the gene frequency back to the dividing frequency 0.5. Furthermore, the higher the mutation rate γ_m, the stronger the force of pullback. These characteristics match what we mentioned in the previous section: Mutation plays an important role in balancing the intensified preference by OB-Scan. In the following sections we will further demonstrate this point theoretically and empirically.

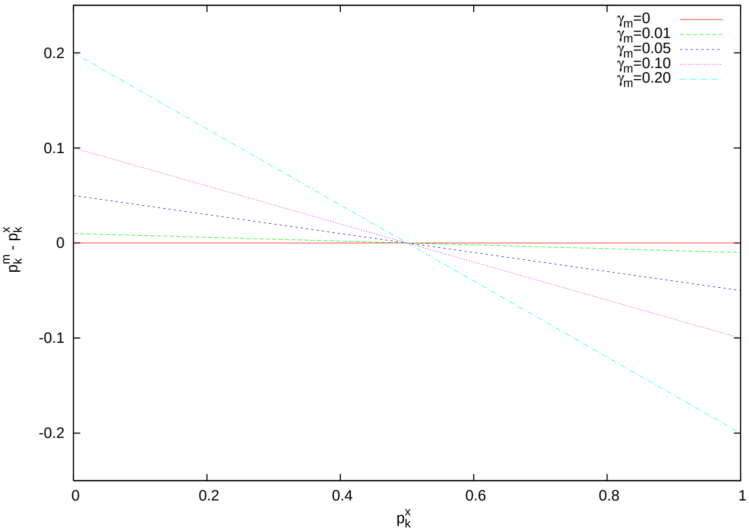


Fig. 4.2. The differential of gene frequency ($p_k^m - p_k^x$) induced by bit-flip mutation with mutation rate γ_m

4.2 Modeling with Markov Chain Theory

Markov chain theory has been used to model the behavior of GAs [5, 13, 50, 56, 64, 100] and to analyze the global convergence of GAs [2, 19, 26, 77, 85, 113]. In this thesis, we use Markov chain theory to formulate the variation of gene frequency in the course of evolution.

4.2.1 Markov Chains

Genetic algorithms are commonly viewed as stochastic processes or stochastic optimization algorithms. A stochastic process is a series of random variables $\{X(t)\}$ over time t. According to the index set type of time t, a stochastic process $\{X(t)\}$ is categorized as a *continuous-time* stochastic process if $t \in [0, \infty)$, or a *discrete-time* stochastic process if $t \in \mathbb{Z}_*$. A large class of stochastic processes have the property that the prediction of value $X(t+1)$ depends only upon the present value $X(t)$ and is independent of the historical values $\{X(0), X(1), \ldots, X(t-1)\}$. Such processes are called *Markov processes*; the memoryless property is known as the *Markov property.* In Markov processes, the possible values of $\{X(t)\}$ are called *states*, and the set of all possible states is called *state space.* In this work we refer to a discrete-time Markov process as a *Markov chain* if the state space is finite or countably infinite. Some well-known definitions about Markov chains [10, 12, 60] are given in Definition 4.13–4.15 below.

Definition 4.13 (Markov Chains). *A stochastic process $\{X(t) : t \in \mathbb{Z}_*\}$ with finite state space S is said to be a* Markov chain *if for all $i_0, i_1, \ldots, i_{t-1}, i, j \in S$*

$$\begin{aligned}\Pr\{X(t+1) = j \mid X(t) = i, X(t-1) = i_{t-1}, \ldots, X(0) = i_0\} \\ = \Pr\{X(t+1) = j \mid X(t) = i\} \\ \overset{\text{def}}{=} \rho_{ij}(t)\ .\end{aligned}$$

The $|S| \times |S|$ square matrix $\mathbf{P}(t) = (\rho_{ij}(t))$ is called the transition matrix *of the Markov chain $\{X(t)\}$. The probability $\rho_{ij}(t)$ is called the* transition probability *of state i to state j at time t.*

Definition 4.14 (Homogeneous Markov Chains). *A Markov chain is said to be* homogeneous *if its transition probabilities are independent of time. That is, for all $t \in \mathbb{Z}_*$*

$$\rho_{ij} = \Pr\{X(t+1) = j \mid X(t) = i\}.$$

A Markov chain consists of a family of random variables; therefore it has a probability distribution of states for each random variable. In terms of Markov processes, this probability distribution is called *state distribution.*

Definition 4.15 (State Distribution).

1. *Denoted by a probability vector $\boldsymbol{\pi}(t) = (\pi_0(t), \ldots, \pi_N(t))$, the* state distribution *of a Markov chain $\{X(t)\}$ with state space $S = \{0, \ldots, N\}$ represents the unconditional probabilities of states. Precisely, for $t \in \mathbb{Z}_*$ and $i \in S$*

$$\pi_i(t) \overset{\text{def}}{=} \Pr\{X(t) = i\} \quad \textit{and} \quad \sum_{i \in S} \pi_i(t) = 1.$$

2. *The vector $\boldsymbol{\pi}(0)$ is called the* initial distribution *of $\{X(t)\}$.*

Next, we introduce a useful theorem for the state distribution of homogeneous Markov chains.

Theorem 4.16 ([12, p.57]). *Suppose we have a homogeneous Markov chain with transition matrix* $\mathbf{P}$. *Given the initial distribution* $\boldsymbol{\pi}(0)$, *the state distribution at time* t *can be obtained by*

$$\boldsymbol{\pi}(t) = \boldsymbol{\pi}(0)\mathbf{P}^t, \tag{4.6}$$

where $\mathbf{P}^t$ *is the* t^{th} *power of matrix* $\mathbf{P}$.

4.2.2 The Model for Gene Frequency

In the light of gene frequency, a GA can be viewed as a stochastic process manipulating the number of allele 1 (or 0) in the population: Let random variables $G_k(t) \in \{0, 1, \dots, m\}$ be the number of allele 1 at locus k at generation t. The process of GAs on the gene frequency can be represented as $\{G_k(t) : t \in \mathbb{Z}_*\}$. Since for every $i_0, i_1, \cdots, i_{t+1} \in \{0, 1, \dots, m\}$ the process $\{G_k(t)\}$ satisfies

$$\begin{aligned}\Pr\{G_k(t+1) = i_{t+1} \mid G_k(t) = i_t, G_k(t-1) = i_{t-1}, \dots, G_k(0) = i_0\} \\ = \Pr\{G_k(t+1) = i_{t+1} \mid G_k(t) = i_t\},\end{aligned}$$

the process $\{G_k(t)\}$ is a Markov chain. A formal definition of the Markov chain for gene frequency is given as follows.

Definition 4.17 (Markov Model for Gene Frequency). *In the Markov chain* $\{G_k(t)\}$ *for the gene frequency at locus* $k \in \{1, \dots, l\}$ *in GAs,*

1. *the* state *is defined as the number of allele* 1 *at locus* k *in the population; thereby the state space is* $\{0, 1, \dots, m\}$. *A state* i *in* $\{G_k(t)\}$ *gives the gene frequency*
$$p_k = \frac{i}{m}.$$
2. *The* transition matrix *of* $\{G_k(t)\}$ *is defined as*
$$\mathbf{P} \overset{\text{def}}{=} \begin{pmatrix} \rho_{00} & \rho_{01} & \cdots & \rho_{0m} \\ \rho_{10} & \rho_{11} & \cdots & \vdots \\ \vdots & \vdots & & \vdots \\ \rho_{m0} & \cdots & \cdots & \rho_{mm} \end{pmatrix},$$
where ρ_{ij} *is the* transition probability *of state* i *to state* j.
$$\rho_{ij} \overset{\text{def}}{=} \Pr\{G_k(t+1) = j \mid G_k(t) = i\}.$$

In the following sections we will utilize this Markov model for gene frequency to investigate the genetic drift and the convergence of MPGAs.

4.3 The Genetic Drift of MPGAs

Genetic drift is a phenomenon that in a finite population without external forces, e.g. selection and mutation, the genetic variability of a locus will decay with time and eventually get fixed to some allele. The cause of genetic drift is the cumulation of sampling errors [40, 53]. In population genetics, the rate of genetic drift serves as an important index of how fast population diversity is lost. Recently this rate is applied to analysis of GAs as a quantitative measure of the losing rate of population diversity [5, 100]. Note that the conditions to examine genetic drift — no selection pressure and no mutation — imply the setting of random selection and zero mutation rate in GAs. The procedure for analysis of genetic drift in GAs is illustrated in Fig. 4.3.

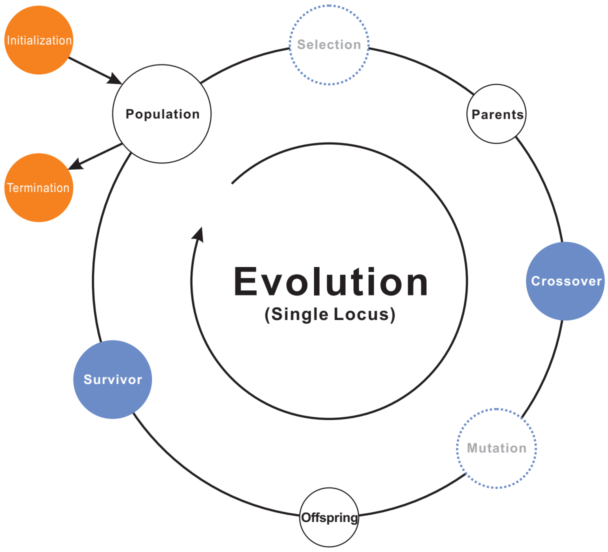


Fig. 4.3. Procedure for analysis of genetic drift in GAs

In the light of MPGAs, Schippers [90] studied the genetic drift of MPGAs using U-Scan and OB-Scan. His work revealed that U-Scan has no influence on genetic drift whilst OB-Scan induces severe genetic drift, as the number of parents is increased. Nevertheless, Schippers only compared the probabilities of drift in and drift out; the rate of genetic drift in MPGAs is still an open question.

In this chapter we investigate in theory the genetic drift of MPGAs using U-Scan and OB-Scan to answer the following questions:

- How long (in generations) will a locus in MPGAs get fixed? Namely, how fast is the genetic drift of MPGAs?
- How likely will this locus get fixed to allele 1 (or 0)?

To this end, we utilize the Markov model for gene frequency presented in Definition 4.17 to derive the mean convergence time — the principal measure of genetic drift in GAs [5].

With regard to the genetic drift for a single locus, the term ‘convergence’ represents the occurrence of all-zeros or all-ones population. In general, random selection and zero mutation rate are assumed for analysis of genetic drift. Therefore the variation of gene frequency in GAs, referring to Definition 4.3, turns out to be:

$$p_k(t) \xrightarrow{\text{crossover}} p_k^x(t) \xrightarrow{\text{survivor}} p_k(t+1). \tag{4.7}$$

Following the above framework and the analytical results in the previous section, we derive the transition probability of the Markov chain for analysis of genetic drift in MPGAs.

Theorem 4.18. *For a GA using random selection, n-parent OB-Scan, and no mutation, the transition probability ρ_{ij} of the Markov chain $\{G_k(t)\}$ corresponding to that GA is*

$$\begin{aligned} \textit{Generational GA:} \quad \rho_{ij} &= B(j; m, p_k') \\ \textit{Steady-state GA:} \quad \rho_{ij} &= p_k' \cdot H(j-1; m-1, i, m) \\ &\quad + (1-p_k') \cdot H(j; m-1, i, m) \end{aligned}$$

with

$$p_k' = I_{\frac{i}{m}}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right).$$

Proof. Recall that the state i of transition probability ρ_{ij} gives the gene frequency $p_k = \frac{i}{m}$. From Lemma 4.7 we can obtain the gene frequency p_k' of the offspring reproduced by a GA using random selection, n-parent OB-Scan, and no mutation. Following the sequence of GAs described in (4.7), the gene frequency is

$$p_k' = p_k^{\text{ob}} = I_{\frac{i}{m}}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right).$$

In generational GAs, the population is completely replaced with the subpopulation consisting of m offspring, which are reproduced by m times of *selection-crossover-mutation* process. Since this process is independent, the number of allele 1 holds a binomial distribution $B(m, p_k')$. Hence, the transition probability is

$$\begin{aligned} \rho_{ij} &= \Pr\{G_k(t+1) = j \mid G_k(t) = i\} \\ &= B(j; m, p_k'). \end{aligned}$$

In steady-state GAs, the subpopulation consists of only one offspring and will replace one chromosome randomly selected from the population at each generation. The probability that the offspring holds allele 1 at locus k is simply p_k'. Moreover, selecting one chromosome out of the population can be viewed as sampling $(m-1)$ members from the population *without replacement* [83]. Thus the number of allele 1 in the remaining population at locus k holds a hypergeometric distribution $H(m-1, i, m)$. Accordingly the transition probability is

$$\rho_{ij} = p_k' \cdot H(j-1; m-1, i, m) + (1-p_k') \cdot H(j; m-1, i, m)$$

□

Theorem 4.19. *For a GA using random selection, n-parent U-Scan, and no mutation, the transition probability ρ_{ij} of the Markov chain $\{G_k(t)\}$ is*

$$\begin{aligned} \textit{Generational:} \quad \rho_{ij} &= B(j; m, \frac{i}{m}) \\ \textit{Steady-state:} \quad \rho_{ij} &= \left(\frac{i}{m}\right) H(j-1; m-1, i, m) \\ &+ \left(1 - \frac{i}{m}\right) H(j; m-1, i, m) \end{aligned}$$

Proof. Lemma 4.4 tells the gene frequency p'_k of the offspring reproduced by a GA using random selection, n-parent U-Scan, and no mutation is

$$p'_k = p^{\mathrm{u}}_k = p_k.$$

The state i of transition probability ρ_{ij} gives the gene frequency $p_k = \frac{i}{m}$. Referring to the proof of Theorem 4.18, we have the transition probability for generational GAs:

$$\rho_{ij} = B(j; m, p'_k) = B(j; m, \frac{i}{m}),$$

and for steady-state GAs:

$$\begin{aligned} \rho_{ij} &= p'_k \cdot H(j-1; m-1, i, m) + (1 - p'_k) \cdot H(j; m-1, i, m) \\ &= \left(\frac{i}{m}\right) H(j-1; m-1, i, m) + \left(1 - \frac{i}{m}\right) H(j; m-1, i, m). \end{aligned}$$

□

Remark 4.20. The transition probability for U-Scan is independent of the number of parents n.

Remark 4.21. Comparing Theorem 4.18 and Theorem 4.19, we learn that the result in Theorem 4.19 is a special case of OB-Scan with $n = 2$. This consequence corresponds to Corollary 4.8. Again, since this correspondence of n-parent U-Scan with 2-parent OB-Scan, in the following text we only discuss OB-Scan.

Proposition 4.22. *The Markov chain $\{G_k(t)\}$ for the GA given in Theorem 4.18 is homogeneous.*

Proof. A Markov chain is said to be homogeneous if the transition probabilities remain constant over time. The transition probability ρ_{ij} of the Markov chain $\{G_k(t)\}$, as shown in Theorem 4.18, is independent of time t. Therefore, the Markov chain $\{G_k(t)\}$ is homogeneous. □

Since the Markov chain $\{G_k(t)\}$ given in Theorem 4.18 is homogeneous, we can utilize (4.6) to compute its state distribution.

4.3.1 Convergence in Genetic Drift

Theorem 4.18 gave the transition matrix of the Markov chain $\{G_k(t)\}$ for the GA using random selection, n-parent OB-Scan, no mutation, and generational or steady-state survivor. Of particular interest to us is, if at all, the *convergence* of $\{G_k(t)\}$ — at that time the population turns out to be all-zeros or all-ones. By means of Markov models, conventional GAs without mutation have been shown to hold such a convergence [50]. In this section we will show that MPGAs without mutation have this convergence property as well. For this, we need to prove that the Markov chains corresponding to the aforementioned MPGAs belong to a special kind of Markov chains, called *absorbing* Markov chains. As implied by the name, this kind of Markov chains will absorb the process into certain states. From the properties of absorbing Markov chains we will derive the mean time and the probability of convergence.

First of all, we draw some well-known definitions (Definitions 4.23–4.25) related to absorbing Markov chains [10, 79].

Definition 4.23 (Absorbing States).

1. *The* closed set S^c *is a set of states whose transition probabilities*

$$\rho_{ij} = 0 \quad \textit{for all } i \in S^c, j \notin S^c.$$

2. *A state i is said to be* absorbing *if and only if*

$$\exists S^c : S^c = \{i\} \Longleftrightarrow \rho_{ii} = 1.$$

3. *A Markov chain with absorbing states is called an absorbing Markov chain.*

Definition 4.24 (First Passage Probability). *Let $f_{ij}^{(t)}$ be the probability that starting from state i, the process for the first time reaches state j in t steps. For a Markov chain $\{X(t)\}$,*

$$f_{ij}^{(t)} \stackrel{\text{def}}{=} \Pr\{X(t) = j, X(v) \neq j \textit{ for } 0 < v < t \mid X(0) = i\}.$$

The first passage probability f_{ij} *is defined as the probability that starting from state i, the process ever reaches state j:*

$$f_{ij} \stackrel{\text{def}}{=} \sum_{t=1}^{\infty} f_{ij}^{(t)}.$$

Definition 4.25 (Transient States). *A state i is said to be* persistent *(or* recurrent*) if $f_{ii} = 1$. Otherwise ($f_{ii} < 1$), the state i is said to be* transient *(or* non-recurrent*).*

According to the above definitions, we prove that the Markov chain associated with the genetic drift of MPGAs is an absorbing Markov chain.

Proposition 4.26. *Suppose we have a GA using random selection, n-parent OB-Scan, no mutation, and either generational or steady-state survivor. The Markov chain $\{G_k(t)\}$ corresponding to this GA is an absorbing Markov chain with exactly two absorbing states:* 0 *and* m.

Proof. According to the definition of absorption,

$$\{G_k(t)\} \text{ is absorbing } \iff \exists i : \rho_{ii} = 1. \tag{4.8}$$

Next, we prove the absorption of $\{G_k(t)\}$ in generational GAs and steady-state GAs individually.

1. Generational GAs: From Theorem 4.18 we know
$$\exists i : \rho_{ii} = 1 \iff \exists i : B(i; m, p'_k) = 1.$$
The solutions of $B(i; m, p'_k) = 1$ subject to $p'_k = I_{\frac{i}{m}}\left(\lceil \frac{n}{2} \rceil, \lceil \frac{n}{2} \rceil\right)$ and $n \in \mathbb{N}_{>1}$ are (i) $i = 0$ with $p'_k = 0$ and (ii) $i = m$ with $p'_k = 1$. This leads to, for the Markov chain $\{G_k(t)\}$,
$$\rho_{00} = \rho_{mm} = 1 \implies \{G_k(t)\} \text{ is absorbing .}$$
2. Steady-state GAs: The transition probability in Theorem 4.18 gives
$$\begin{aligned}\rho_{ii} &= p'_k \cdot H(i-1; m-1, i, m) + (1 - p'_k) \cdot H(i; m-1, i, m) \\ &= p'_k \cdot \frac{\binom{i}{i-1}\binom{m-i}{m-i}}{\binom{m}{m-1}} + (1 - p'_k) \cdot \frac{\binom{i}{i}\binom{m-i}{m-1-i}}{\binom{m}{m-1}} \\ &= p'_k \cdot \frac{i}{m} + (1 - p'_k) \cdot \frac{m-i}{m} \\ &= 1 - \frac{i}{m} - \left(1 - \frac{2i}{m}\right) I_{\frac{i}{m}}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right)\end{aligned}$$
The definition of absorption implies
$$\exists i : \rho_{ii} = 1 \iff \exists i : (1 - \frac{i}{m}) - \left(1 - \frac{2i}{m}\right) I_{\frac{i}{m}}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right) = 1.$$
For all $n \in \mathbb{N}_{>1}$ and $m \in \mathbb{N}$, the solutions of the above equation are $i = 0$ and $i = m$. Therefore, for the Markov chain $\{G_k(t)\}$ of steady-state GAs, we have
$$\rho_{00} = \rho_{mm} = 1 \implies \{G_k(t)\} \text{ is absorbing .}$$

Concluding the cases in generational GAs and in steady-state GAs, we complete the proof that the Markov chain $\{G_k(t)\}$ is absorbing with exactly two absorbing states 0 and m. □

With respect to absorbing Markov chains, of interest to us are the time and the probability to get absorbed into the absorbing states. The *absorption probability* of an absorbing state is the probability that the process starts from a transient state and eventually gets absorbed into that absorbing state. Recall that the first passage probability f_{ij} represents the probability that starting from state i, the process ever reaches state j. Thus, the probability f_{ij} represents the absorption probability for an absorbing state j and a transient state i.

To compute the absorption probability and the mean time to absorption, we introduce the fundamental matrix [12] in Definition 4.27 and its related properties in Theorem 4.28.

Definition 4.27 (Fundamental Matrix). *For a Markov chain with b absorbing states, the transition matrix can be rewritten as*

$$\mathbf{P} = \begin{pmatrix} \mathbf{I}_b & \mathbf{0} \\ \mathbf{R} & \mathbf{Q} \end{pmatrix}, \tag{4.9}$$

where $\mathbf{I}_b$ is a $b \times b$ identity matrix. Denoting by $\mathbf{I}$ the identity matrix, the fundamental matrix *for the absorbing Markov chain is defined as*

$$\mathbf{F} \overset{\text{def}}{=} (\mathbf{I} - \mathbf{Q})^{-1}.$$

Theorem 4.28 ([12, p.155]). *Of an absorbing Markov chain, let $\mathbf{F}$ be the fundamental matrix and $\mathbf{R}$ be the matrix described in the preceding definition.*

1. *The components of $\mathbf{FR} = (f_{ij})$ represent the absorption probability f_{ij} of transient state i to absorbing state j.*
2. *The fundamental matrix $\mathbf{F} = (\tau_{ij})$ stands for the mean time τ_{ij} that the process spends at transient state j assuming it starts from transient state i.*

Using Theorem 4.28, we derive the mean convergence time of MPGAs without mutation, that is, the rate of genetic drift.

Theorem 4.29 (Mean Convergence Time of MPGAs without Mutation). *Suppose we have a GA using random selection, n-parent OB-Scan, no mutation, and either generational or steady-state survivor. Let $\mathbf{F} = (\tau_{ij})$ be the fundamental matrix of the Markov chain $\{G_k(t)\}$ corresponding to this GA. For locus $k \in \{1, \ldots, l\}$, given its initial state distribution $\boldsymbol{\pi}(0) = (\pi_0(0), \ldots, \pi_m(0))$, the mean convergence time is*

$$\tau = \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} \pi_i(0)\tau_{ij}.$$

Proof. We have shown that the Markov chain $\{G_k(t)\}$ has two absorbing states 0 and m corresponding to the predefined GA. Let A be the set of absorbing states in $\{G_k(t)\}$, namely $A \equiv \{0, m\}$. Theorem 4.28 indicates the fundamental matrix of the Markov chain $\{G_k(t)\}$ representing the mean time that the process spends at state $j \in \bar{A}$ starting from state $i \in \bar{A}$. The mean time τ_i that the process spends among transient states, assuming it starts from transient state i, can be derived:

$$\tau_i = \sum_{j \in \bar{A}} \tau_{ij}.$$

Therefore, given the initial state distribution $\boldsymbol{\pi}(0)$, we have the mean convergence time

$$\tau = \sum_{i=1}^{m-1} \tau_i \Pr(i \mid t = 0) = \sum_{i=1}^{m-1} \tau_i \pi_i(0) = \sum_{i=1}^{m-1} \sum_{j=1}^{m-1} \pi_i(0) \tau_{ij}.$$

□

Theorem 4.30 (Probability of Convergence). *Let $\{G_k(t)\}$ be the Markov chain corresponding to the GA using random selection, n-parent OB-Scan, and no mutation. For locus $k \in \{1, \ldots, l\}$, given the initial state distribution $\boldsymbol{\pi}(0) = (\pi_0(0), \ldots, \pi_m(0))$, the probabilities of convergence to all-zeros and all-ones population are respectively*

$$\lim_{t \to \infty} \Pr\{\forall \mathbf{c} \in C : c_k = 0\} = \pi_0(0) + \sum_{i=1}^{m-1} \pi_i(0) \cdot (\mathbf{FR})_{i0},$$
$$\lim_{t \to \infty} \Pr\{\forall \mathbf{c} \in C : c_k = 1\} = \pi_m(0) + \sum_{i=1}^{m-1} \pi_i(0) \cdot (\mathbf{FR})_{im},$$

where $(\mathbf{FR})_{ij}$ denote the (i,j) element of the matrix product $\mathbf{FR}$.

Proof. Since $\{G_k(t)\}$ is an absorbing Markov chain with two absorbing states 0 and m, the absorption probabilities

$$f_{i0} = \lim_{t \to \infty} \Pr\{G_k(t) = 0 \mid G_k(0) = i\},$$
$$f_{im} = \lim_{t \to \infty} \Pr\{G_k(t) = m \mid G_k(0) = i\}.$$

Theorem 4.28 tells the absorption probability of transient state $i \in \{1, \ldots, m-1\}$ to absorbing state $j \in \{0, m\}$:

$$f_{ij} = (\mathbf{FR})_{ij}.$$

Additionally, the absorption probability for $i, j \in \{0, m\}$ is

$$f_{ij} = \delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

Hence the probability for the GA to converge to all-zeros population is

$$\begin{aligned}\lim_{t\to\infty} \Pr\{\forall \mathbf{c} \in C : c_k = 0\} &= \lim_{t\to\infty} \Pr\{G_k(t) = 0\} \\ &= \lim_{t\to\infty} \sum_{i=0}^{m} \Pr\{G_k(t) = 0 \mid G_k(0) = i\} \Pr\{G_k(0) = i\} \\ &= \sum_{i=0}^{m} f_{i0}\pi_i(0) \\ &= \pi_0(0) + \sum_{i=1}^{m-1} \pi_i(0) \cdot (\mathbf{FR})_{i0}\end{aligned}$$

The probability of convergence to all-ones population can be proved similarly. □

Example 4.31. Consider the locus k of a generational GA using population size 6, random selection, 3-parent OB-Scan, and no mutation. According to Theorem 4.18, the transition matrix of the corresponding Markov chain is

$$\mathbf{P} = \begin{array}{c} \\ 0\\1\\2\\3\\4\\5\\6 \end{array}\!\!\begin{array}{c} \begin{array}{ccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \left(\begin{array}{ccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0.63 & 0.30 & 0.06 & 6.5\text{e-}3 & 3.9\text{e-}4 & 1.2\text{e-}5 & 1.7\text{e-}7 \\ 0.17 & 0.35 & 0.30 & 0.14 & 0.04 & 5.2\text{e-}3 & 3.0\text{e-}4 \\ 0.02 & 0.09 & .023 & 0.31 & 0.23 & 0.09 & 0.02 \\ 3.0\text{e-}4 & 5.2\text{e-}3 & 0.04 & 0.14 & 0.30 & 0.35 & 0.17 \\ 1.7\text{e-}7 & 1.2\text{e-}5 & 3.9\text{e-}4 & 6.5\text{e-}3 & 0.06 & 0.30 & 0.63 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right) \end{array} .$$

Obviously, the states 0 and 6 are absorbing states; thus this chain is absorbing. Following Definition 4.27, the transition matrix can be rewritten as

$$\mathbf{P} = \begin{array}{c} \\ 0\\6\\1\\2\\3\\4\\5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccccc} 0 & 6 & 1 & 2 & 3 & 4 & 5 \end{array} \\ \left(\begin{array}{ccccccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0.63 & 1.7\text{e-}7 & 0.30 & 0.06 & 6.5\text{e-}3 & 3.9\text{e-}4 & 1.2\text{e-}5 \\ 0.17 & 3.0\text{e-}4 & 0.35 & 0.30 & 0.14 & 0.04 & 5.2\text{e-}3 \\ 0.02 & 0.02 & 0.09 & .023 & 0.31 & 0.23 & 0.09 \\ 3.0\text{e-}4 & 0.17 & 5.2\text{e-}3 & 0.04 & 0.14 & 0.30 & 0.35 \\ 1.7\text{e-}7 & 0.63 & 1.2\text{e-}5 & 3.9\text{e-}4 & 6.5\text{e-}3 & 0.06 & 0.30 \end{array}\right) \end{array}$$

with

$$\mathbf{R} = \begin{pmatrix} 0.63 & 1.7\text{e-}7 \\ 0.17 & 3.0\text{e-}4 \\ 0.02 & 0.02 \\ 3.0\text{e-}4 & 0.17 \\ 1.7\text{e-}7 & 0.63 \end{pmatrix}, \text{ and } \mathbf{Q} = \begin{pmatrix} 0.30 & 0.06 & 6.5\text{e-}3 & 3.9\text{e-}4 & 1.2\text{e-}5 \\ 0.35 & 0.30 & 0.14 & 0.04 & 5.2\text{e-}3 \\ 0.09 & .023 & 0.31 & 0.23 & 0.09 \\ 5.2\text{e-}3 & 0.04 & 0.14 & 0.30 & 0.35 \\ 1.2\text{e-}5 & 3.9\text{e-}4 & 6.5\text{e-}3 & 0.06 & 0.30 \end{pmatrix} .$$

The mean time τ_{ij} for $i, j \in \{1, \ldots, 5\}$ can then be obtained from the fundamental matrix

$$(\tau_{ij}) = \mathbf{F} = (\mathbf{I} - \mathbf{Q})^{-1} = \begin{pmatrix} 1.51 & 0.15 & 0.05 & 0.03 & 0.02 \\ 0.88 & 1.66 & 0.40 & 0.24 & 0.19 \\ 0.57 & 0.67 & 1.74 & 0.67 & 0.57 \\ 0.19 & 0.24 & 0.40 & 1.66 & 0.88 \\ 0.02 & 0.03 & 0.05 & 0.15 & 1.51 \end{pmatrix},$$

and the absorption probabilities f_{ij} for $i \in \{1, \ldots, 5\}$, $j \in \{0, 6\}$

$$(f_{ij}) = \mathbf{FR} = \begin{pmatrix} 0.981 & 0.019 \\ 0.836 & 0.164 \\ 0.5 & 0.5 \\ 0.164 & 0.836 \\ 0.019 & 0.981 \end{pmatrix}.$$

Assume the population is initialized randomly. The elements of the initial distribution are

$$\pi_i(0) = B\left(i; 6, \frac{1}{2}\right) \quad \text{for } i = 0, 1, \ldots, 6,$$

and therefore the initial distribution is

$$\boldsymbol{\pi}(0) = \left(\frac{1}{64}, \frac{6}{64}, \frac{15}{64}, \frac{20}{64}, \frac{15}{64}, \frac{6}{64}, \frac{1}{64}\right).$$

The mean convergence time is derived by

$$\tau = \left(\frac{6}{64}, \frac{15}{64}, \frac{20}{64}, \frac{15}{64}, \frac{6}{64}\right) \cdot \begin{pmatrix} 1.51 + 0.15 + 0.05 + 0.03 + 0.02 \\ 0.88 + 1.66 + 0.40 + 0.24 + 0.19 \\ 0.57 + 0.67 + 1.74 + 0.67 + 0.57 \\ 0.19 + 0.24 + 0.40 + 1.66 + 0.88 \\ 0.02 + 0.03 + 0.05 + 0.15 + 1.51 \end{pmatrix} = 3.23 \ .$$

Moreover, the probability of convergence to all-ones population is

$$\lim_{t \to \infty} \Pr\{\forall \mathbf{c} \in C : c_k = 1\} = \frac{1}{64} + \left(\frac{6}{64}, \frac{15}{64}, \frac{20}{64}, \frac{15}{64}, \frac{6}{64}\right) \cdot \begin{pmatrix} 0.019 \\ 0.164 \\ 0.5 \\ 0.836 \\ 0.981 \end{pmatrix} = 0.5 \ .$$

More theoretical results of the mean time and the probability of convergence to allele 1 are presented in the following section.

4.3.2 Theoretical Results and Experimental Validation

This section demonstrates the theoretical results obtained from the above theorems with regard to the mean time and the probability of convergence. Moreover, we conduct experiments on a single locus ($l = 1$) to verify these results. The setting of MPGAs used in our experiments is generational GA, bit-string representation, random selection, and no mutation. Each experiment setting includes 1000 independent runs.

4.3.2.1 Generational MPGAs

First we check the results of MPGAs using OB-Scan. Figure 4.4 depicts the mean convergence time for n parents and population size m. First, this figure shows that our theoretical results fit the experimental results very well. Second, it shows that for $n \in 2\mathbb{N}_{>1}$ a GA using n-parent OB-Scan performs correspondingly to that using (n–1)-parent OB-Scan, which confirms the *pairwise equivalence* claimed in Corollary 4.9. Third, this figure indicates that using more than two parents in OB-Scan causes a drastic decrease in mean convergence time, compared to the use of two parents in OB-Scan. Nonetheless, raising parents from more than two parents only yields a small decrease. The mean convergence time for $m = 256$ and $n = 2$, for example, amounts to 351.55 generations while it takes only 10.17 generations for $n = 3$ (or 4) and 6.94 generations for $n = 5$ (or 6). This speedup in convergence reflects that OB-Scan with more than two parents accelerates genetic drift.

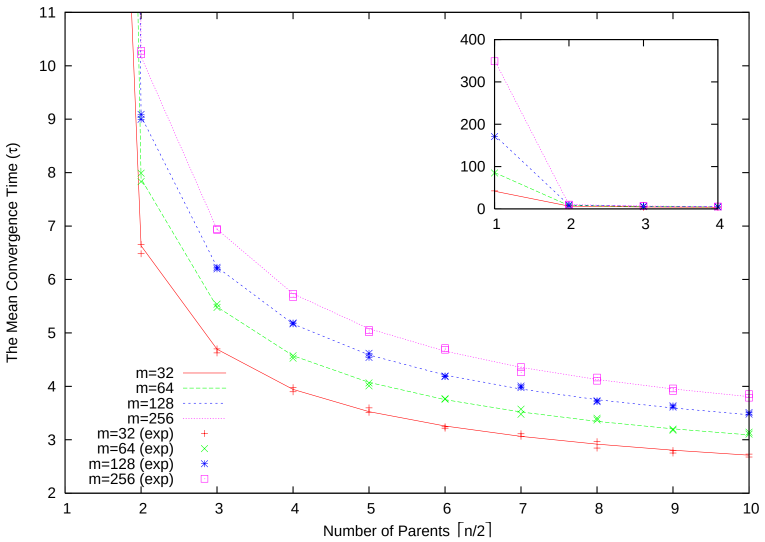


Fig. 4.4. The mean convergence time τ for n-parent OB-Scan and population size m (the lines representing the theoretical results and the symbols representing the experimental results)

The above results are obtained from random initialization, namely, the expected initial gene frequency being $\frac{1}{2}$. On the mean convergence time, the influence of the initial gene frequency is further examined. We denote by β the bias of initial gene frequency to the allele 1. Figure 4.5 shows the mean convergence time for the bias $\beta = \frac{1}{32}$ and $\beta = \frac{1}{8}$. Similar to random initialization, applying bias at initialization also has the tendency that raising parents will reduce the mean convergence time, especially from two to three parents. Concerning the effect of bias, the small bias ($\frac{1}{32}$) slightly decreases the mean convergence time, compared to no bias (i.e. random initialization) shown in Fig. 4.4. However, the bias $\beta = \frac{1}{8}$ substantially reduces the mean convergence time for more than two parents. In other words, it further accelerates genetic drift. Additionally, the difference between different population sizes is relatively little in mean convergence time for $\beta = \frac{1}{8}$.

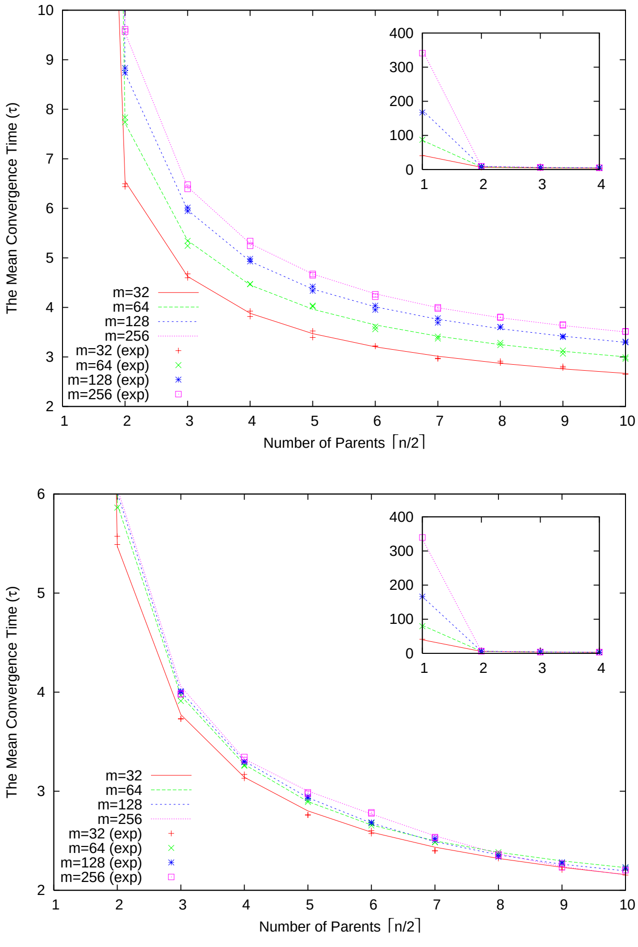


Fig. 4.5. The mean convergence time τ for n-parent OB-Scan and population size m with initialization bias $\beta = \frac{1}{32}$ (top) and $\beta = \frac{1}{8}$ (bottom)

Figure 4.6 compares the progress of genetic drift of uniform crossover (i.e. 2-parent OB-Scan) and 3-parent OB-Scan for population size $m = 32$ under initialization bias $\beta = \frac{1}{32}$. The rate of genetic drift for 3-parent OB-Scan is much faster than that for uniform crossover. Interestingly, the distribution of convergence probability of uniform crossover differs from that of 3-parent OB-Scan either. Asoh and Mühlenbein [5] have shown the convergence probability of uniform crossover equals the initialization probability, which is reflected in Fig. 4.6. Yet, adopting more parents does not follow this rule. The 3-parent OB-Scan gives a probability (≈ 0.6) higher than the initialization probability ($\frac{17}{32} \approx 0.531$). This outcome implies the number of parents in OB-Scan exerts an influence upon the probabilities of convergence to all-zeros and all-ones population. To investigate this influence, we examine the convergence probability for different numbers of parents and population sizes under initialization bias β. Figure 4.7 shows the probabilities of convergence for initialization bias $\beta = \frac{1}{32}$ and $\beta = \frac{1}{8}$. Obviously, using more than two parents has a higher probability of convergence to the allele 1: the larger the initialization bias, the higher the probability. Raising the number of parents further increases this probability. In a word, using more parents will intensify the preference of the initialization. However, the increment of preference has its limit. As shown in Fig. 4.7, it receives no significant increase in the probability of convergence as the number of parents is more than 10.

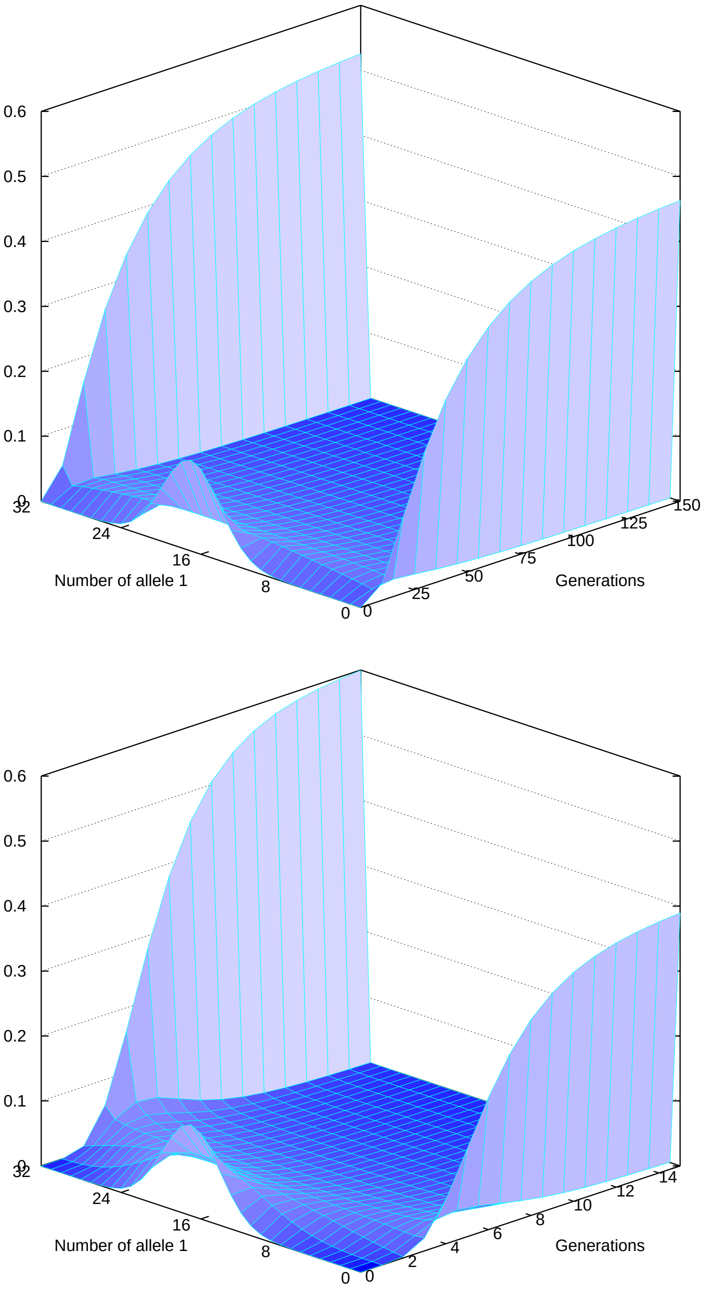


Fig. 4.6. The progress of genetic drift of 2-parent OB-Scan (top) and 3-parent OB-Scan (bottom) for population size $m = 32$ with initialization bias $\beta = \frac{1}{32}$

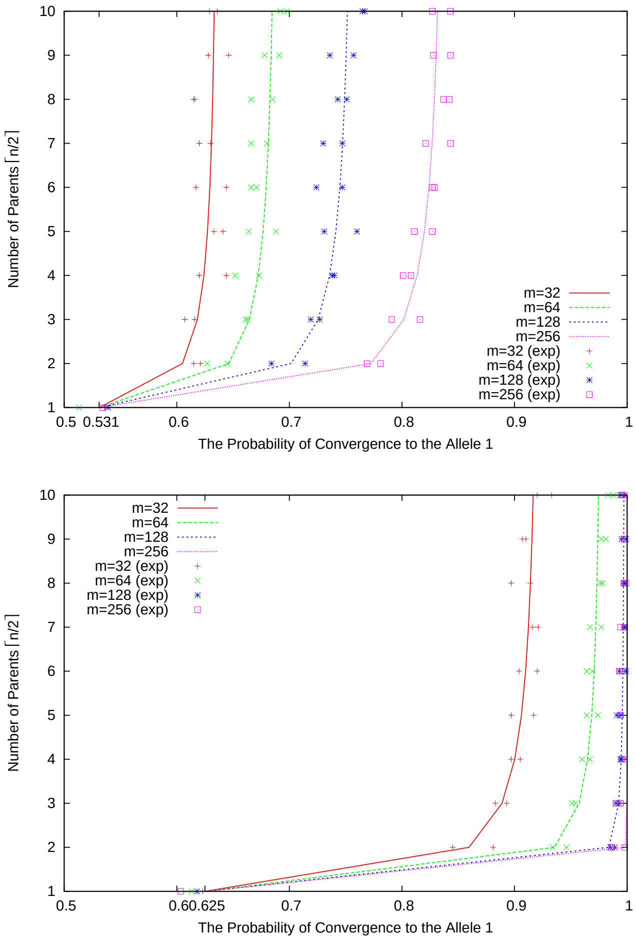


Fig. 4.7. The probability of convergence to all-ones population for the GA using n-parent OB-Scan and initialization bias $\beta = \frac{1}{32}$ (top) and $\beta = \frac{1}{8}$ (bottom)

Next, we examine the convergence of MPGAs using U-Scan. The mean convergence time, as shown in Fig. 4.8, does not vary significantly with the number of parents for all the four sizes of population. In addition, Fig. 4.9 demonstrates that the difference in convergence probability between U-Scan using different numbers of parents is insignificant: the probabilities all stand round the initialization probability. This invariance with different numbers of parents in U-Scan becomes clearer, compared to the results of OB-Scan in Fig. 4.4 and Fig. 4.7. These outcomes validate our argument: The number of parents in U-Scan has no influence upon the gene frequency, and further the mean time and probability of convergence. Recall that both 2-parent U-Scan and 2-parent OB-Scan correspond to uniform crossover. It is sufficient to say that U-Scan with any number of parents performs identically with uniform crossover in terms of the effect on genetic drift.

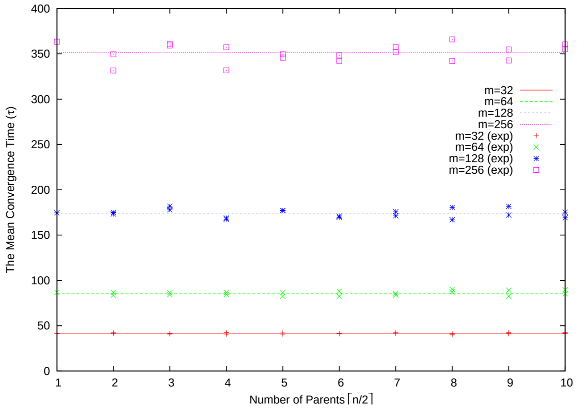


Fig. 4.8. The mean convergence time τ for n-parent U-Scan and population size m (the lines representing the theoretical results and the symbols representing the experimental results)

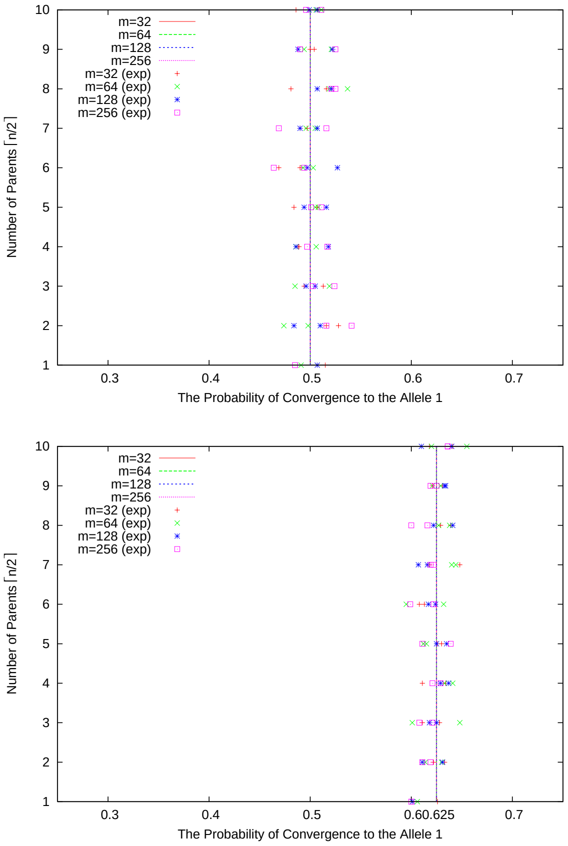


Fig. 4.9. The probability of convergence to all-ones population for the GA using n-parent U-Scan and initialization bias $\beta = 0$ (top) and $\beta = \frac{1}{8}$ (bottom)

4.3.2.2 Steady-State MPGAs

This section examines the genetic drift in steady-state MPGAs using OB-Scan. Figure 4.10 compares the theoretical and the experimental mean convergence time of MPGAs using OB-Scan under no initialization bias ($\beta = 0$) and $\beta = \frac{1}{8}$ for different sizes of population. Figure 4.11 further depicts the probability for the MPGA to converge to allele 1. The good fit between the theoretical and the experimental results in these figures confirms the capability of the proposed Markov model for the genetic drift in steady-state MPGAs. In addition, the tendencies of steady-state MPGAs in mean convergence time and probability are very similar to those of generational MPGAs: First, the pairwise equivalence also holds in steady-state MPGAs using OB-Scan. Second, the mean convergence time increases drastically from the number of parents $n = 2$ to $n = 3$ (or 4), but then increases gradually from $n = 3$ (or 4) to $n = 5$ (or 6). Third, raising parents in OB-Scan gains a higher probability of convergence than the initialization bias to allele 1.

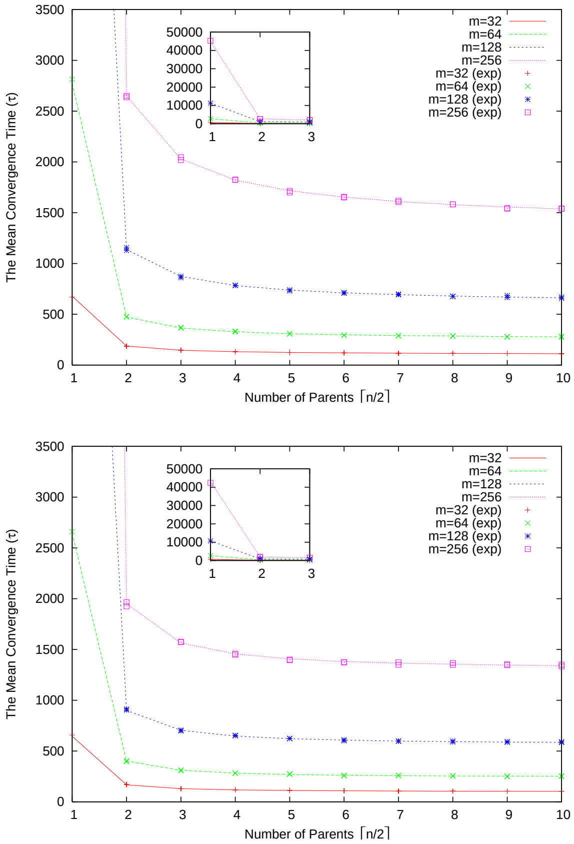


Fig. 4.10. The mean convergence time τ for the steady-state MPGA using n-parent OB-Scan and population size m with initialization bias $\beta = 0$ (top) and $\beta = \frac{1}{8}$ (bottom)

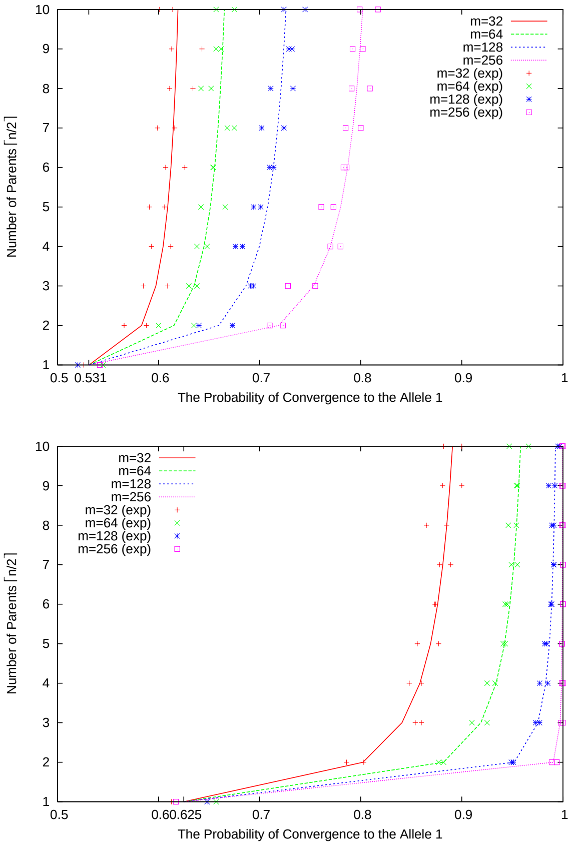


Fig. 4.11. The probability of convergence to all-ones population for the steady-state MPGA using n-parent OB-Scan and initialization bias $\beta = \frac{1}{32}$ (top) and $\beta = \frac{1}{8}$ (bottom)

We further compare the mean convergence time and probability of steady-state MPGAs with those of generational MPGAs. Here the number of generations in steady-state MPGAs is divided by population size m for an equivalent number of fitness evaluations to generational MPGAs in one generation. Figures 4.12 and 4.13 respectively compare the mean convergence time and probability of generational and steady-state MPGAs using OB-Scan. These two figures show that steady-state MPGAs have longer mean convergence time and lower probability of convergence to allele 1 than generational MPGAs. In addition, the number of generations reduced by adding parents in steady-state MPGAs is smaller than that in generational MPGAs; however, the difference between them is small, precisely, not more than 5 generations except for $n = 2$. In a word, the number of parents in OB-Scan has a stronger influence over generational MPGAs than over steady-state MPGAs in terms of genetic drift, to wit, the mean convergence time and probability.

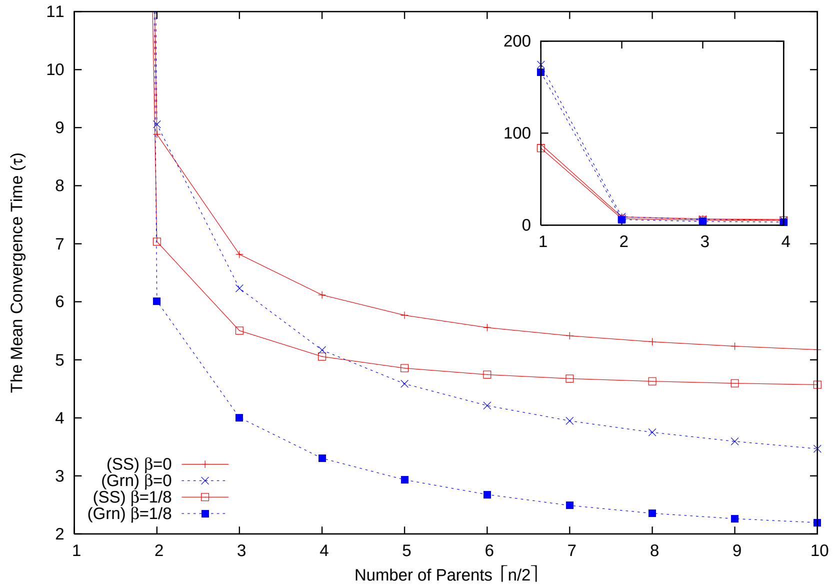


Fig. 4.12. Comparison between the generational MPGA (Grn) and the steady-state MPGA (SS) using OB-Scan in mean convergence time for population size $m = 128$

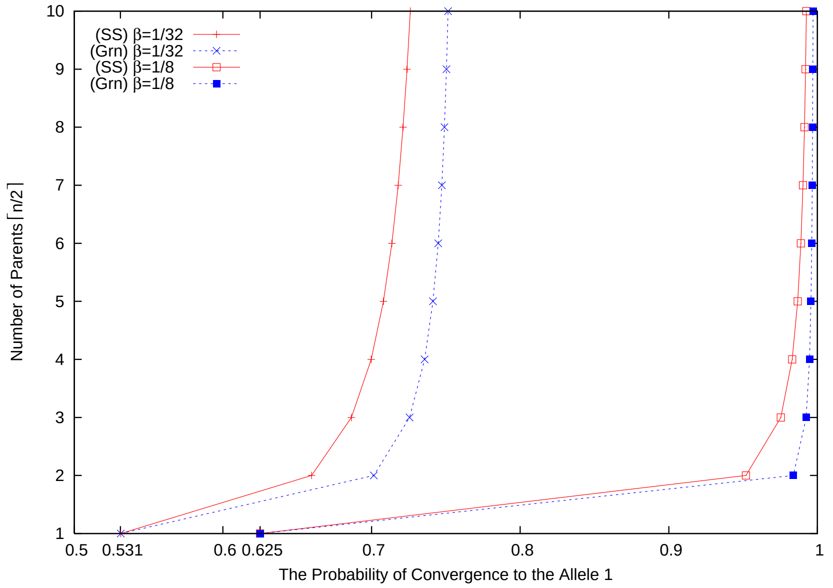


Fig. 4.13. Comparison between the generational MPGA (Grn) and the steady-state MPGA (SS) using OB-Scan in probability of convergence to allele 1 for population size $m = 128$

4.4 The Convergence of MPGAs

This section investigates the influence of raising parents on the convergence of MPGAs using U-Scan and OB-Scan. In this section, the analysis will focus on the OneMax problem domain. This problem is widely used in the analysis of evolutionary algorithms since it supports several useful probabilistic properties and the building block hypothesis [48]. Even though the OneMax problem is relatively simple, the analysis on it founds a basic understanding of how the (MP)GA operators work and interact in the course of evolution. Furthermore, the analytical models for the OneMax problems are promisingly applicable to other problem domains [68, 89].

In Section 4.1 we analyzed the variation of gene frequency affected by crossover and mutation. Here we further explore the impact of selection on the gene frequency in the OneMax problem. Using the Markov model for gene frequency proposed in Section 4.2, we model the variation of gene frequency over time for the evolution of MPGAs. The procedure for our analysis on the convergence of MPGAs is illustrated in Fig. 4.14. Additionally, the mean convergence fitness and time for MPGAs in the OneMax problem will be derived. Through this Markov model we examine in theory the separate as well as the integral influences of *population size*, *selection intensity*, *the number of parents*, *mutation rate*, and *generation gap* on the convergence speed and solution quality of MPGAs. Accordingly we are able to deal with the following questions:

- When or in which situations will MPGAs outperform GAs, concerning selection intensity, the number of parents, mutation rate, and generation gap?
- What is the best number of parents in the OneMax problem?

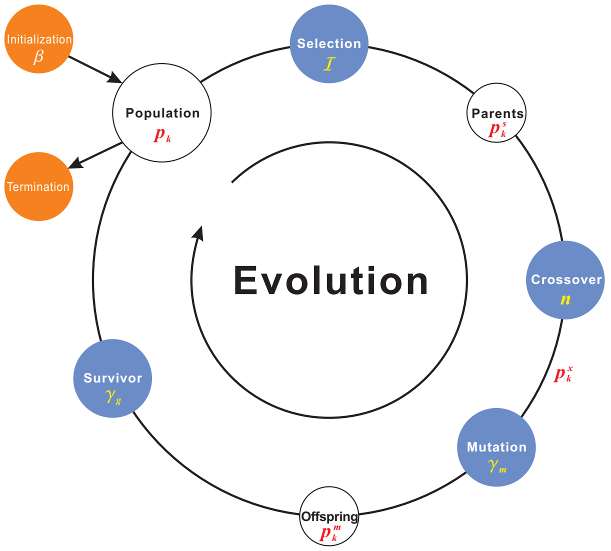


Fig. 4.14. Procedure for analysis of convergence of MPGAs

4.4.1 Background: The OneMax Problem

Recall the fact that most of the selection operators in GAs depend on fitness — a *phenotypic* property of chromosomes. Conversely, gene frequency concerns the *genotypes* of chromosomes. The mapping from genotypes to phenotype is problem-dependent in essence. Thus the analysis associated with selection depends on the problem domain. In this section, our analysis focuses on the OneMax problem. This problem domain is commonly used for theoretical analysis of evolutionary algorithms because of the following reasons. First, the fitness function of the OneMax problem provides a direct way to bridge the gap between phenotype and genotype. Second, even though the OneMax problem is relatively simple, the theoretical analysis for it provides an insight into how selection interacts with crossover, mutation and survivor, and how these operators affect the overall performance in theory. Furthermore, the analysis for the OneMax problem is promisingly applicable to other problem domains [68, 89].

The definition of the OneMax problem and its related properties are given as follows.

Definition 4.32 (The OneMax Problem). *The OneMax problem is to find the binary string* $\mathbf{c} = (c_1, \ldots, c_l) \in \{0,1\}^l$ *that maximizes the following function* $f : \{0,1\}^l \rightarrow \{0, 1, \ldots, l\}$*:*

$$f(\mathbf{c}) = \sum_{k=1}^{l} c_k.$$

The OneMax problem has the following properties:

1. The contribution of gene to fitness is identical and independent.
2. The genes are independent of each other.
3. It supports the building block hypothesis.

The first property is clear since in the OneMax problem the contribution of allele 1 to fitness is exactly one (zero for allele 0), regardless of the locus. In addition, the genes at different loci are independent of each other. For the third property, due to the length of building blocks in the OneMax problem being one, crossover will not disrupt the built blocks. This characteristic supports the building block hypothesis, even for uniform crossover [99].

The independence of genes in the OneMax problem leads to the following lemma.

Lemma 4.33. *In the OneMax problem, the covariance of the genes at any two loci* $u, v \in \{1, \ldots, l\}$ *is*

$$\mathrm{Cov}(c_u, c_v) = 0.$$

Proof. Since the genes are independent of each other in the OneMax problem, the expectation $\mathrm{E}[c_u c_v] = \mathrm{E}[c_u]\mathrm{E}[c_v]$. Therefore, it holds for the covariance that

$$\mathrm{Cov}(c_u, c_v) = \mathrm{E}[c_u c_v] - \mathrm{E}[c_u]\mathrm{E}[c_v] = 0.$$

□

4.4.2 Variation of Gene Frequency Caused by Selection

Definition 4.3 expressed the process of GAs on the gene frequency as

$$p_k(t) \xrightarrow{\text{selection}} p_k^s(t) \xrightarrow{\text{crossover}} p_k^x(t) \xrightarrow{\text{mutation}} p_k^m(t) \xrightarrow{\text{survivor}} p_k(t+1).$$

In Section 4.1 we have derived the variation of gene frequency following the above process, except the variation caused by selection. In this section we will deal with the remaining part: $p_k(t) \xrightarrow{\text{selection}} p_k^s(t)$. Consequently, all the influences of MPGA operators on the gene frequency are acquired.

Before conducting the analysis, we give a definition of mean fitness, which will be extensively discussed later on. The relation between mean fitness and the gene frequency will be presented afterwards.

Definition 4.34 (Mean Fitness). *The* mean fitness *of a population C is defined as*

$$\bar{f} = \frac{1}{|C|} \sum_{\mathbf{c} \in C} f(\mathbf{c}).$$

Lemma 4.35. *Let p_k be the gene frequency at locus k. In the OneMax problem the mean $\bar{f}$ and the variance σ_F^2 of fitness are*

$$\bar{f} = \sum_{k=1}^{l} p_k$$

$$\sigma_F^2 = \sum_{k=1}^{l} \sigma_k^2 = \sum_{k=1}^{l} p_k(1-p_k)$$

Proof. The mean fitness can be obtained by (4.1):

$$\bar{f} = \mathrm{E}[f] = \mathrm{E}\left[\sum_{k=1}^{l} c_k\right] = \sum_{k=1}^{l} \mathrm{E}\left[c_k\right] = \sum_{k=1}^{l} p_k.$$

According to Lemma 4.33, we can derive the variance of fitness by

$$\begin{aligned} \sigma_F^2 = \mathrm{Var}(f) = \mathrm{Var}\left(\sum_{k=1}^{l} c_k\right) &= \sum_{k=1}^{l} \mathrm{Var}(c_k) + \sum_{u \neq v} \mathrm{Cov}(c_u, c_v) \\ &= \sum_{k=1}^{l} \sigma_k^2 = \sum_{k=1}^{l} p_k(1-p_k). \end{aligned}$$

□

Remark 4.36. Similarly, let C^s be the set of selected parents, we have the gene frequency of allele 1 among the selected parents

$$p_k^s = \frac{1}{|C^s|} \sum_{\mathbf{c} \in C^s} c_k,$$

and the mean fitness of the selected parents

$$\bar{f}^s = \frac{1}{|C^s|} \sum_{\mathbf{c} \in C^s} f(\mathbf{c}) = \sum_{k=1}^{l} p_k^s.$$

To compute the variation of gene frequency caused by selection, here we introduce the concept of *selection intensity*, which originates from quantitative genetics [40] but now is commonly used in the evolutionary computation (EC) community as a quantitative measure of selection pressure [11, 71, 72, 100]. The definitions of selection differential and selection intensity are given as follows.

Definition 4.37 (Selection Differential). *Let $\bar{f}(t)$ and $\bar{f}^s(t)$ be the mean fitness of the population and that of selected parents at generation t, respectively. The* selection differential *$S(t)$ is defined as*

$$S(t) = \bar{f}^s(t) - \bar{f}(t).$$

Definition 4.38 (Selection Intensity). *Let $S(t)$ be the selection differential and $\sigma_F^2(t)$ be the variance of fitness in the population at generation t. The* selection intensity *is defined as*

$$\mathcal{I}(t) = \frac{S(t)}{\sigma_F(t)} = \frac{\bar{f}^s(t) - \bar{f}(t)}{\sigma_F(t)}. \tag{4.10}$$

As above-stated, this selection intensity $\mathcal{I}$ serves as a quantitative measure in the influence of selection on fitness. However, we need another measure to examine the impact of selection on the gene frequency. In this work, a *selection intensity for gene frequency* is additionally defined in order to measure the effect of selection upon the gene frequency.

Definition 4.39 (Selection Intensity for Gene Frequency). *The selection intensity $\mathcal{I}_k^{\mathrm{p}}$ for gene frequency at locus $k \in \{1, \ldots, l\}$ is defined as*

$$\mathcal{I}_k^{\mathrm{p}} \overset{\mathrm{def}}{=} \frac{p_k^s - p_k}{\sigma_k},$$

where σ_k is the standard deviation, defined as the square root of the variance σ_k^2.

Lemma 4.40. *In the OneMax problem, the expected selection intensity for gene frequency is*

$$\mathrm{E}\left[\mathcal{I}^{\mathrm{p}}\right] = \frac{\mathcal{I}}{\sqrt{l}} \sqrt{1 + \frac{\mathrm{Var}(\sigma)}{\mathrm{E}^2[\sigma]}}. \tag{4.11}$$

Proof. The definition of selection intensity for gene frequency tells

$$p_k^s - p_k = \mathcal{I}_k^{\mathrm{p}} \sigma_k .$$

Since $\mathcal{I}_k^{\mathrm{p}}$ and σ_k are independent, averaging all loci gives

$$\mathrm{E}[p^s] - \mathrm{E}[p] = \mathrm{E}\left[\mathcal{I}^{\mathrm{p}}\right] \mathrm{E}[\sigma]. \tag{4.12}$$

According to Lemma 4.35 and (4.10), the above equation can be rewritten as

$$\begin{aligned}\mathrm{E}\left[\mathcal{I}^{\mathrm{p}}\right] &= \frac{\mathrm{E}[p^s] - \mathrm{E}[p]}{\mathrm{E}[\sigma]} = \frac{\frac{1}{l}\sum_{k=1}^{l} p_k^s - \frac{1}{l}\sum_{k=1}^{l} p_k}{\mathrm{E}[\sigma]} = \frac{\bar{f}^s - \bar{f}}{l\mathrm{E}[\sigma]} \\ &= \frac{\mathcal{I}\sigma_F}{l\mathrm{E}[\sigma]} = \frac{\mathcal{I}\sqrt{\sum_{k=1}^{l}\sigma_k^2}}{l\mathrm{E}[\sigma]} = \frac{\mathcal{I}\sqrt{l\mathrm{E}[\sigma^2]}}{l\mathrm{E}[\sigma]} = \frac{\mathcal{I}}{\sqrt{l}}\sqrt{\frac{\mathrm{E}[\sigma^2]}{\mathrm{E}^2[\sigma]}}.\end{aligned}$$

The variance $\mathrm{Var}(\sigma) = \mathrm{E}[\sigma^2] - \mathrm{E}^2[\sigma]$. Thus,

$$\mathrm{E}\left[\mathcal{I}^{\mathrm{p}}\right] = \frac{\mathcal{I}}{\sqrt{l}}\sqrt{\frac{\mathrm{E}^2[\sigma] + \mathrm{Var}(\sigma)}{\mathrm{E}^2[\sigma]}} = \frac{\mathcal{I}}{\sqrt{l}}\sqrt{1 + \frac{\mathrm{Var}(\sigma)}{\mathrm{E}^2[\sigma]}}.$$

□

Remark 4.41. As $\mathrm{Var}(\sigma) \ll \mathrm{E}^2[\sigma]$, the expected selection intensity for gene frequency is

$$\mathrm{E}\left[\mathcal{I}^{\mathrm{p}}\right] \approx \frac{\mathcal{I}}{\sqrt{l}} .$$

Here we utilize the above expectation as the selection intensity for gene frequency at any locus k. Consequently, the variation of gene frequency caused by selection approximates to

$$p_k^s \approx p_k + \left(\frac{\mathcal{I}}{\sqrt{l}}\right) \sigma_k . \tag{4.13}$$

A practical advantage of (4.13) is its correlation with selection intensity $\mathcal{I}$. Accordingly, we can utilize the existing analytical results, e.g. [11], on the selection intensity $\mathcal{I}$ of diverse selection operators in GAs. Note that the selection intensity for gene frequency in (4.13) is based on the assumption $\mathrm{Var}(\sigma) \ll \mathrm{E}^2[\sigma]$. Even though this assumption does not necessarily hold in the course of evolution, we will empirically show this approximation can work well in Section 4.4.5.

In addition, equation (4.13) indicates that a positive selection intensity will result in a gene frequency p_k^s higher than the gene frequency p_k. The stronger the selection intensity, the higher the increment of gene frequency. Figure 4.15 plots the differential of gene frequency induced by selection in the 100-bit OneMax problem, i.e. $l = 100$. Obviously, raising selection intensity increases the differential of gene frequencies between p_k^s and p_k. In the OneMax problem it means that raising selection intensity will always improve the mean fitness of selected parents. On the other hand, a long chromosome length will lower the increment of gene frequency. Precisely, the increment of gene frequency, as shown in (4.13), is in inverse proportion to the square root of l.

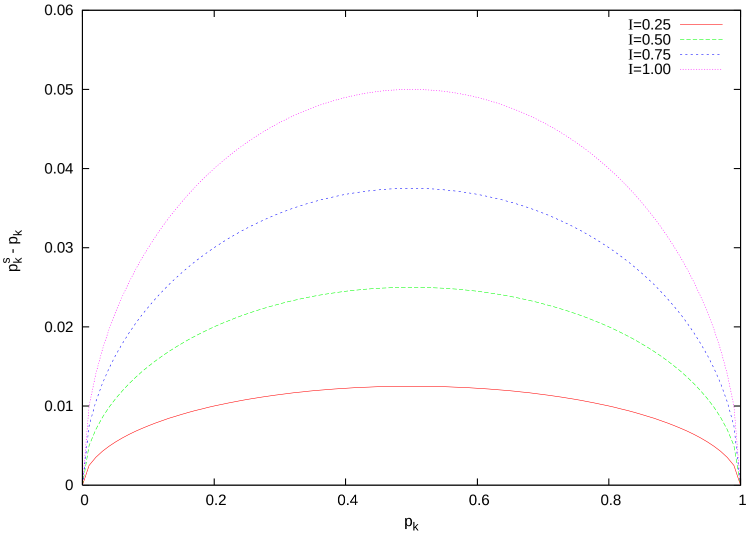


Fig. 4.15. The differential of gene frequency $(p_k^s - p_k)$ induced by performing selection with selection intensity $\mathcal{I}$ on p_k

4.4.3 The Markov Model for MPGAs

Beyond separate influence, the integral influence of MPGA operators upon the performance is considered in this section. To this end, we follow the Markov model proposed in Definition 4.17 to formulate the variation of gene frequency over generations. Moreover, in the following analysis we will prove that the utilization of mutation in MPGAs leads to an *ergodic* Markov chain instead of an *absorbing* chain. From this Markov model, the mean fitness at generation t and the mean convergence time in the OneMax problem will be derived.

Before presenting the Markov model, we give the definition of survivor and generation gap adopted in this thesis.

Definition 4.42 (Survivor and Generation Gap). *Subject to the generation gap* $\gamma_g \overset{\text{def}}{=} \frac{\lambda}{m}$ *with* $\lambda \in \{1, \ldots, m\}$*, the survivor randomly selects* λ *chromosomes from the population and replaces them with* λ *offspring.*

With reference to the Markov model for gene frequency $\{G_k(t)\}$ presented in Definition 4.17, we derive below its transition probabilities considering the integral influence of selection, crossover, mutation, and survivor.

Theorem 4.43. *For a GA using selection with selection intensity* $\mathcal{I}$*, n-parent OB-Scan, bit-flip mutation with mutation* γ_m*, and survivor with generation gap* $\gamma_g = \frac{\lambda}{m}$*, the transition probability* ρ_{ij} *of the Markov chain* $\{G_k(t)\}$ *corresponding to that GA is*

$$\rho_{ij} = \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) \cdot H(j - x; m - \lambda, i, m)$$

with

$$p_k' = I_{\frac{i}{m}+\mathcal{I}\sqrt{\frac{i(m-i)}{m^2 l}}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) + \gamma_m\left(1 - 2I_{\frac{i}{m}+\mathcal{I}\sqrt{\frac{i(m-i)}{m^2 l}}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)\right).$$

Proof. The starting state i of transition probability ρ_{ij} gives the gene frequency $p_k = \frac{i}{m}$. From (4.13), Lemma 4.7, and Lemma 4.12, we can compute the gene frequency p_k' of the offspring reproduced by the predefined GA:

$$\begin{aligned}
p_k' &= p_k^x + \gamma_m\left(1 - 2p_k^x\right) \\
&= I_{p_k^s}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) + \gamma_m\left(1 - 2I_{p_k^s}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)\right) \\
&= I_{p_k+\frac{\mathcal{I}}{\sqrt{l}}\sqrt{p_k(1-p_k)}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) \\
&\quad + \gamma_m\left(1 - 2I_{p_k+\frac{\mathcal{I}}{\sqrt{l}}\sqrt{p_k(1-p_k)}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)\right) \\
&= I_{\frac{i}{m}+\mathcal{I}\sqrt{\frac{i(m-i)}{m^2 l}}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) + \gamma_m\left(1 - 2I_{\frac{i}{m}+\mathcal{I}\sqrt{\frac{i(m-i)}{m^2 l}}}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)\right).
\end{aligned}$$

Regarding generation gap $\gamma_g = \frac{\lambda}{m}$, there will be λ out of m chromosomes randomly selected from the population and replaced with λ offspring. According to Rogers and Prügel-Bennett's manner [83], the resulting population can be regarded as a union of a subset A consisting of λ offspring and a subset B formed by selecting $(m - \lambda)$ members at random from the population without replacement. Denote by $|A_1|$ and $|B_1|$ the number of allele 1 in A and B, respectively. Since offspring are reproduced *independently* by λ times of *selection-crossover-mutation* process, the number $|A_1|$ holds a binomial distribution $B(\lambda, p_k')$. Hence, the p.m.f.

$$\Pr\{|A_1| = x \mid G_k(t) = i\} = B(x; \lambda, p_k').$$

Furthermore, the number of allele 1 in B holds a hypergeometric distribution $H(m - \lambda, i, m)$ because of random selection without replacement. Thus,

$$\Pr\{|B_1| = y \mid G_k(t) = i\} = H(y; m - \lambda, i, m).$$

Since A and B are independent, the transition probability is

$$\begin{aligned}
\rho_{ij} &= \Pr\{G_k(t+1) = j \mid G_k(t) = i\} \\
&= \sum_{x+y=j} \Pr\{|A_1| = x, |B_1| = y \mid G_k(t) = i\} \\
&= \sum_{x=0}^{\lambda} \Pr\{|A_1| = x \mid G_k(t) = i\} \Pr\{|B_1| = j - x \mid G_k(t) = i\} \\
&= \sum_{x=0}^{\lambda} B(x; \lambda, p_k') \cdot H(j - x; m - \lambda, i, m).
\end{aligned}$$

□

Remark 4.44. For generational GAs (generation gap $\gamma_g = 1$) and steady-state GAs (generation gap $\gamma_g = \frac{1}{m}$), the above theorem gives the transition probabilities of the corresponding Markov chains:

$$\begin{aligned} \text{Generational GAs:} \quad & \rho_{ij} = B(j; m, p'_k) \\ \text{Steady-state GAs:} \quad & \rho_{ij} = (1 - p'_k) \cdot H(j; m-1, i, m) \\ & \qquad + p'_k \cdot H(j-1; m-1, i, m) \end{aligned}$$

Remark 4.45. According to Corollary 4.8, for the predefined GA but using U-Scan, the transition probability is equal to that of OB-Scan with $n = 2$ in Theorem 4.43.

Proposition 4.46. *The Markov chain $\{G_k(t)\}$ corresponding to the GA given in Theorem 4.43 is homogeneous.*

Proof. A Markov chain is said to be homogeneous if the transition probabilities remain constant over time. The transition probability ρ_{ij} of the Markov chain $\{G_k(t)\}$, as shown in Theorem 4.43, is independent of time t. Therefore, the Markov chain $\{G_k(t)\}$ is homogeneous. □

Since the Markov chain $\{G_k(t)\}$ given in Theorem 4.43 is homogeneous, we can utilize (4.6) to compute the state distribution at generation t. This leads to the following theorem.

Theorem 4.47 (Mean Fitness). *Suppose we have the state distribution $\boldsymbol{\pi}(t) = (\pi_0(t), \ldots, \pi_m(t))$ corresponding to the GA given in Theorem 4.43. In the OneMax problem, the mean $\bar{f}(t)$ and the variance $\sigma_F^2(t)$ of fitness are*

$$\bar{f}(t) = \frac{l}{m} \sum_{j=0}^{m} j\pi_j(t)$$

$$\sigma_F^2(t) = \frac{l}{m^2} \sum_{j=0}^{m} j(m-j)\pi_j(t)$$

Proof. Given the state distribution $\boldsymbol{\pi}(t)$ for locus $k \in \{1, \ldots, l\}$ at generation t, we have the gene frequency $p_k(t)$

$$p_k(t) = \sum_{j=0}^{m} \left(\frac{j}{m}\right) \Pr\{G_k(t) = j\} = \frac{1}{m} \sum_{j=0}^{m} j\pi_j(t).$$

With the above equation, from Lemma 4.35 we obtain the mean fitness at generation t

$$\bar{f}(t) = \sum_{k=1}^{l} p_k(t) = \frac{l}{m} \sum_{j=0}^{m} j\pi_j(t)$$

and the variance

$$\sigma_F^2(t) = \sum_{k=1}^{l} \left(p_k(t) - p_k^2(t)\right) = l \sum_{j=0}^{m} \left(\frac{j}{m} - \left(\frac{j}{m}\right)^2\right) \pi_j(t) = \frac{l}{m^2} \sum_{j=0}^{m} j(m-j)\pi_j(t).$$

□

4.4.4 Convergence

Theorem 4.43 gave the transition matrix of the Markov chain $\{G_k(t)\}$ for the GA using selection with selection intensity $\mathcal{I}$, n-parent OB-Scan, bit-flip mutation with mutation rate γ_m, and survivor with generation gap γ_g. In this section, we look further into the *convergence* of $\{G_k(t)\}$ — at that time the state distribution will stay invariant once the process achieves it. In terms of Markov chains [12], such a state distribution is referred to as *stationary distribution* (or *equilibrium distribution*) and this chain is called *stationary.* We will show that the MPGA given in Theorem 4.43 has this convergence property. Furthermore, the mean convergence fitness and time in the OneMax problem will be derived.

Lemma 4.48 (Ergodicity [10, p.50]). *A Markov chain is* ergodic[1] *if and only if its transition matrix is primitive, i.e.*

$$\exists t \in \mathbb{Z}_+ : \mathbf{P}^t > \mathbf{0},$$

where $\mathbf{0}$ *denotes* zero matrix *and* $\mathbf{P}^t > \mathbf{0}$ *represents that all components in* $\mathbf{P}^t$ *are strictly positive.*

Theorem 4.49 (Stationary Distribution [10, p.50]). *For an ergodic Markov chain with state space* $\{0, 1, \ldots, m\}$*,*

1. *the powers* $\mathbf{P}^t$ *of the transition matrix* $\mathbf{P}$ *converge componentwise to a matrix whose all rows are equal. If we denote a typical row by* $\boldsymbol{\pi} = (\pi_0, \ldots, \pi_m)$*, then we have* $\pi_i > 0$ *and* $\sum_i \pi_i = 1$ *for all* $i \in \{0, \ldots, m\}$*.*
2. $\boldsymbol{\pi}$ *is the unique vector such that*
$$\boldsymbol{\pi}\mathbf{P} = \boldsymbol{\pi}. \tag{4.14}$$

This unique $\boldsymbol{\pi}$ *is called the* stationary distribution *associated with the chain.*

It is worthy of note that the stationary distribution $\boldsymbol{\pi}$ is independent of the initial distribution $\boldsymbol{\pi}(0)$. In other words, the process with any given initial distribution will converge to the same distribution, viz the stationary distribution. In addition, the stationary distribution represents a fixed status, i.e. convergence, of gene frequency. Yet, an issue arises: Does there exist such a stationary distribution in the Markov chain $\{G_k(t)\}$? To prove the existence, we need to prove this chain is ergodic.

Lemma 4.50 ([57, p.517]). *If an* $n \times n$ *matrix* $\mathbf{A}$ *is nonnegative and irreducible, and if all the main diagonal entries of* $\mathbf{A}$ *are positive, then* $\mathbf{A}^{n-1} > \mathbf{0}$*.*

Proposition 4.51. *With mutation rate* $0 < \gamma_m < 1$*, the transition matrix* $\mathbf{P}$ *corresponding to the GA given in Theorem 4.43 is irreducible.*

Proof. The matrix $\mathbf{P}$ is irreducible if the entries $\rho_{i,i-1} > 0$ for all $0 < i \leq m$ and $\rho_{i,i+1} > 0$ for all $0 \leq i < m$. We individually prove these two inequalities as follows.

[1] The term *ergodic* is also referred to as *irreducible and aperiodic.*

1. For $\rho_{i,i-1}$ with $0 < i \leq m$: From Theorem 4.43, we have the transition probability

$$\rho_{i,i-1} = \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) H(i-1-x; m-\lambda, i, m).$$

with

$$p'_k = p^x_k + \gamma_m(1 - 2p^x_k).$$

Since $0 < \gamma_m < 1$ and $0 \leq p^x_k \leq 1$ (Probability Axiom), the probability p'_k is

$$0 < p'_k < 1.$$

This gives the binomial distribution

$$B(x; \lambda, p'_k) > 0. \tag{4.15}$$

For the p.m.f. of hypergeometric distribution, we have

$$\begin{aligned} H(i-1-x; m-\lambda, i, m) > 0 &\iff i \geq i-1-x \text{ and } i-(i-1-x) \leq m-(m-\lambda) \\ &\iff -1 \leq x \leq \lambda - 1 \end{aligned}$$

Therefore,

$$H(i-1-x; m-\lambda, i, m) \begin{cases} > 0 & \text{if } -1 \leq x \leq \lambda - 1, \\ = 0 & \text{otherwise.} \end{cases} \tag{4.16}$$

According to (4.15) and (4.16), it holds for the transition probability with $\lambda \in \{1, \dots, m\}$ that

$$\rho_{i,i-1} = \sum_{x=0}^{\lambda-1} B(x; \lambda, p'_k) H(i-1-x; m-\lambda, i, m) > 0. \tag{4.17}$$

2. For $\rho_{i,i+1}$ with $0 \leq i < m$: Theorem 4.43 gives the transition probability

$$\rho_{i,i+1} = \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) H(i+1-x; m-\lambda, i, m).$$

Similarly, (4.15) holds in this case. For hypergeometric distribution, the p.m.f.

$$\begin{aligned} H(i+1-x; m-\lambda, i, m) > 0 &\iff i \geq i+1-x \text{ and } i-(i+1-x) \leq m-(m-\lambda) \\ &\iff 1 \leq x \leq \lambda + 1 \end{aligned}$$

Hence we have

$$H(i+1-x; m-\lambda, i, m) \begin{cases} > 0 & \text{if } 1 \leq x \leq \lambda + 1, \\ = 0 & \text{otherwise.} \end{cases} \tag{4.18}$$

According to (4.15) and (4.18), it holds for the transition probability with $\lambda \in \{1, \dots, m\}$ that

$$\rho_{i,i+1} = \sum_{x=1}^{\lambda} B(x; \lambda, p'_k) H(i + 1 - x; m - \lambda, i, m) > 0. \tag{4.19}$$

Equations (4.17) and (4.19) give $\rho_{i,i-1} > 0$ for $0 < i \leq m$ and $\rho_{i,i+1} > 0$ for $0 \leq i < m$. Therefore, the transition matrix $\mathbf{P}$ is irreducible. □

Proposition 4.52. *With mutation rate $0 < \gamma_m < 1$, the Markov chain corresponding to the GA given in Theorem 4.43 is ergodic.*

Proof. Theorem 4.43 gives the diagonal entries of transition matrix $\mathbf{P}$ corresponding to the predefined GA:

$$\rho_{ii} = \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) H(i - x; m - \lambda, i, m). \tag{4.20}$$

Equation (4.15) indicates $B(x; \lambda, p'_k) > 0$ for $0 < \gamma_m < 1$. The hypergeometric part of (4.20) is

$$\begin{aligned} H(i - x; m - \lambda, i, m) > 0 &\iff i \geq i - x \text{ and } i - (i - x) \leq \lambda \\ &\iff 0 \leq x \leq \lambda \end{aligned}$$

Since $B(x; \lambda, p'_k) > 0$ and $H(i - x; m - \lambda, i, m) > 0$ for $0 \leq x \leq \lambda$, from (4.20) we have the diagonal entries

$$\rho_{ii} > 0.$$

In addition, the transition matrix $\mathbf{P}$ is defined to be nonnegative and Proposition 4.51 shows this transition matrix is irreducible. Accordingly, Lemma 4.50 gives $\mathbf{P}^m > \mathbf{0}$, and Lemma 4.48 further indicates the ergodicity. □

Lemma 4.53. *Let $\{G_k(t)\}$ be the Markov chain corresponding to the GA given in Theorem 4.43 at locus $k \in \{1, \dots, l\}$. For the chain $\{G_k(t)\}$ there exists the stationary distribution $\boldsymbol{\pi} = (\pi_1, \dots, \pi_l)$, which gives*

$$\lim_{t \to \infty} \Pr\{G_k(t) = j\} = \pi_j.$$

Proof. Proposition 4.52 shows the Markov chain $\{G_k(t)\}$ is ergodic. Referring to Theorem 4.49, there exists the unique stationary distribution $\boldsymbol{\pi}$ giving for all $j \in \{0, \dots, m\}$

$$\pi_j = \lim_{t \to \infty} \Pr\{G_k(t) = j\}.$$

□

Theorem 4.54 (Mean Convergence Fitness). *In the OneMax problem, for the mean $\bar{f}^*$ and variance $\sigma_F^{2^*}$ of convergence fitness of the GA given in Theorem 4.43, we have*

$$\bar{f}^* = \frac{l}{m} \sum_{j=0}^{m} j\pi_j$$

$$\sigma_F^{2^*} = \frac{l}{m^2} \sum_{j=0}^{m} j(m-j)\pi_j$$

Proof. Substitute stationary distribution $\boldsymbol{\pi}$ given in Theorem 4.53 into the equations in Theorem 4.47; then we complete the proof. □

In addition to stationary distribution, we investigate the mean time for the process of GA to achieve these states. To this end, we introduce the fundamental matrix for ergodic chains [12] in Definition 4.55. The mean time can then be derived from the fundamental matrix.

Definition 4.55 (Fundamental Matrix of Ergodic Chains). *For an ergodic Markov chain, the fundamental matrix*

$$\mathbf{Z} \overset{\text{def}}{=} (\mathbf{I} - (\mathbf{P} - \boldsymbol{\Pi}))^{-1},$$

where

$$\boldsymbol{\Pi} = \begin{pmatrix} \boldsymbol{\pi} \\ \boldsymbol{\pi} \\ \vdots \\ \boldsymbol{\pi} \end{pmatrix} = \begin{pmatrix} \pi_0 & \cdots & \pi_m \\ \pi_0 & \cdots & \pi_m \\ \vdots & & \vdots \\ \pi_0 & \cdots & \pi_m \end{pmatrix}.$$

Theorem 4.56 ([12, p.230]). *Let μ_{ij} be the mean time to state j starting from state i. The matrix $\mathbf{M} \overset{\text{def}}{=} (\mu_{ij})$ is given by*

$$\mathbf{M} = (\mathbf{I} - \mathbf{Z} + \mathbf{1}\text{diag}(\mathbf{Z}))\,\text{diag}(\boldsymbol{\Pi})^{-1},$$

where $\mathbf{1}$ represents the matrix with all entries equal to 1 and $\text{diag}(\mathbf{Z})$ is the diagonal matrix which has the same diagonal as $\mathbf{Z}$. Alternatively, the above equation can be explicitly written as[2]

$$\mu_{ij} = (\delta_{ij} - z_{ij} + z_{jj})/\pi_j.$$

Now we launch the derivation of the mean time for the process to achieve stationarity.

Lemma 4.57. *In the fundamental matrix $\mathbf{Z}$, for any row i the sum of entries in this row is one; that is,*

$$\sum_j z_{ij} = 1.$$

[2] Refer to Theorem 7.7 [10, p.55]

Proof. The fundamental matrix can be represented as[3]

$$\mathbf{Z} = \mathbf{I} + \sum_{t\geq 1} \left(\mathbf{P}^t - \mathbf{\Pi}\right).$$

For any t, each row in the matrices $\mathbf{P}^t$ and $\mathbf{\Pi}$ satisfies

$$\sum_j \rho_{ij}^{(t)} = 1 \quad \text{and} \quad \sum_j \pi_j = 1.$$

Hence the sum of entries in any row i of the fundamental matrix

$$\sum_j z_{ij} = \sum_j \left(\delta_{ij} + \sum_{t\geq 1} \left(\rho_{ij}^{(t)} - \pi_j\right)\right) = 1 + \sum_{t\geq 1} \sum_j \left(\rho_{ij}^{(t)} - \pi_j\right) = 1.$$

□

Theorem 4.58 (Mean Convergence Time). *Let $\{G_k(t)\}$ be the Markov chain corresponding to the GA given in Theorem 4.43 at locus $k \in \{1, \ldots, l\}$ and let $\mathbf{Z}$ be the fundamental matrix of $\{G_k(t)\}$. The mean convergence time μ is given by*

$$\mu = \sum \operatorname{diag}(\mathbf{Z}) = \sum_{j=0}^{m} z_{jj}.$$

Proof. Proposition 4.52 has shown $\{G_k(t)\}$ is ergodic. According to Definition 4.55 and Theorem 4.56, the mean time μ_{ij} to state j starting from state i for $i, j \in \{0, \ldots, m\}$ is

$$\mu_{ij} = \left(\delta_{ij} - z_{ij} + z_{jj}\right)/\pi_j.$$

Let μ_i be the mean time starting from state i. Given the stationary distribution $\boldsymbol{\pi}$, we have

$$\begin{aligned} \mu_i &= \sum_{j=0}^{m} \mu_{ij}\pi_j = \sum_{j=0}^{m} \left(\delta_{ij} - z_{ij} + z_{jj}\right) \\ &= 1 - \sum_{j=0}^{m} z_{ij} + \sum_{j=0}^{m} z_{jj}. \end{aligned} \tag{4.21}$$

Lemma 4.57 shows for any i the sum $\sum_{j=0}^{m} z_{ij} = 1$. Thus (4.21) turns into

$$\mu_i = 1 - 1 + \sum_{j=0}^{m} z_{jj} = \sum_{j=0}^{m} z_{jj}. \tag{4.22}$$

[3] Refer to [12, p.226]

Since $\sum_i \pi_i(0) = 1$, the mean convergence time μ is obtained by

$$\mu = \sum_{i=0}^{m} \mu_i \pi_i(0) = \left(\sum_{j=0}^{m} z_{jj} \right) \sum_{i=0}^{m} \pi_i(0) = \sum_{j=0}^{m} z_{jj}.$$

□

Remark 4.59. Equation (4.22) reveals that the mean time to equilibrium μ_i is independent of the starting state i. That is to say, the initial distribution has no effect on the mean time μ_i — this property corresponds to the ineffectiveness of initial distribution on stationarity.

This section built a Markov model to formulate the variation of gene frequency in the course of evolution. The proposed model concerns the effects of selection intensity $\mathcal{I}$ in selection, the number of parents n in multi-parent crossover, mutation rate γ_m in mutation, and generation gap γ_g in survivor. This model demonstrates the influence of these factors on the mean fitness for the OneMax problem. Furthermore, the mean convergence fitness and time are derived from this Markov model. More theoretical results are presented in the next section.

4.4.5 Theoretical Results and Experimental Validation

This section demonstrates theoretical results obtained from the above theorems. Moreover, we conduct a series of experiments on the OneMax problem to validate these theoretical arguments. The setting of MPGAs used in our experiments is listed in Fig. 4.16. The size of the OneMax problem is set to be 100 bits. Each experiment setting includes 100 independent runs.

GA type	Generational GA
Representation	Bit string ($l = 100$)
Population size	128
Selection	Linear ranking selection
Selection bias η^+	1.5
Crossover	U-Scan and OB-Scan with $n = 2, \ldots, 20$
Crossover rate γ_x	1.0
Mutation	Bit-flip mutation
Mutation rate γ_m	0.01, 0.05, 0.1, and 0.2
Termination	1000 generations

Fig. 4.16. The setting of MPGAs in experiments

4.4.5.1 Impact of Selection and Mutation

First of all, we examine MPGAs *without* selection, i.e. the selection intensity $\mathcal{I}$ is zero. This zero selection intensity removes the potential error caused by adopting the expectation as the real selection intensity for gene frequency in (4.13). Accordingly, the model proposed in Theorem 4.43 becomes an exact model. Note that zero selection intensity yields a random selection with no preference for allele 0 or 1. The expected gene frequency will then stay at 0.5, which cannot afford to examine the effects of crossover and mutation. For this we turn to initialize the chromosomes with a predetermined bias. We denote by β the initialization bias in favor of allele 1. Figure 4.17 compares the theoretical and the experimental mean fitness for the MPGA using initialization bias $\beta = \frac{1}{32}$ with mutation rate $\gamma_m = 0.01$ and 0.1 respectively. The good fit between theoretical and experimental results demonstrates the effectiveness of our proposed model. In Fig. 4.17, the profiles of U-Scan stay constant for all numbers of parents. That is, U-Scan with any number of parents corresponds to 2-parent U-Scan and 2-parent OB-Scan. This phenomenon validates Corollary 4.5 and Remark 4.45: The number of parents has no influence on the performance of U-Scan. In other words, n-parent U-Scan with $n \in \mathbb{N}_{>1}$ performs analogously to 2-parent OB-Scan. Moreover, Fig. 4.17 shows that, without selection pressure the mean fitness of 2-parent OB-Scan (viz uniform crossover) is much worse than that of n-parent OB-Scan with $n > 2$. Precisely, 2-parent OB-Scan converges to the fitness $l(0.5 + \beta)$ while using more parents results in higher fitness. This improvement, as mentioned in Section 4.1.2, is attributed to the effect of intensification in OB-Scan. As for the effect of mutation, the increase of mutation rate not only enlarges the variance of convergent fitness but also counteracts the intensification caused by raising parents in OB-Scan. The latter is demonstrated by the deterioration of fitness as mutation rate is increased from 0.01 to 0.1.

Next, we consider MPGAs *with* selection, specifically, GAs using selection with selection intensity $\mathcal{I} > 0$, n-parent OB-Scan, bit-flip mutation with $0 < \gamma_m < 1$, and survivor with generation gap γ_g. In this section we adopt linear ranking selection [52, 114] as the selection operator. The linear ranking selection, as its name tells, assigns the probability for a chromosome to be selected according to a linear formula based on the rank of this chromosome. Let $i \in \{1, \ldots, m\}$ be the rank (1 for the worst chromosome and m for the best one), the probability

$$\Pr\{\mathbf{c}_i \text{ to be selected}\} = \frac{1}{m}\left[\eta^- + (\eta^+ - \eta^-)\frac{i-1}{m-1}\right],$$

where $1 \leq \eta^+ \leq 2$ and $\eta^- = 2 - \eta^+$ are two parameters used to control the linear relation [6]. Precisely, $\frac{\eta^-}{m}$ and $\frac{\eta^+}{m}$ are the expected probabilities for the worst and the best chromosome to be selected, respectively. For the linear ranking selection, Blickle and Thiele [11] derived its selection intensity. In the following text we will use their formula to compute the selection intensity of linear ranking selection:

$$\mathcal{I}_{\mathrm{R}} = (1 - \eta^-)\frac{1}{\sqrt{\pi}} = (\eta^+ - 1)\frac{1}{\sqrt{\pi}}. \tag{4.23}$$

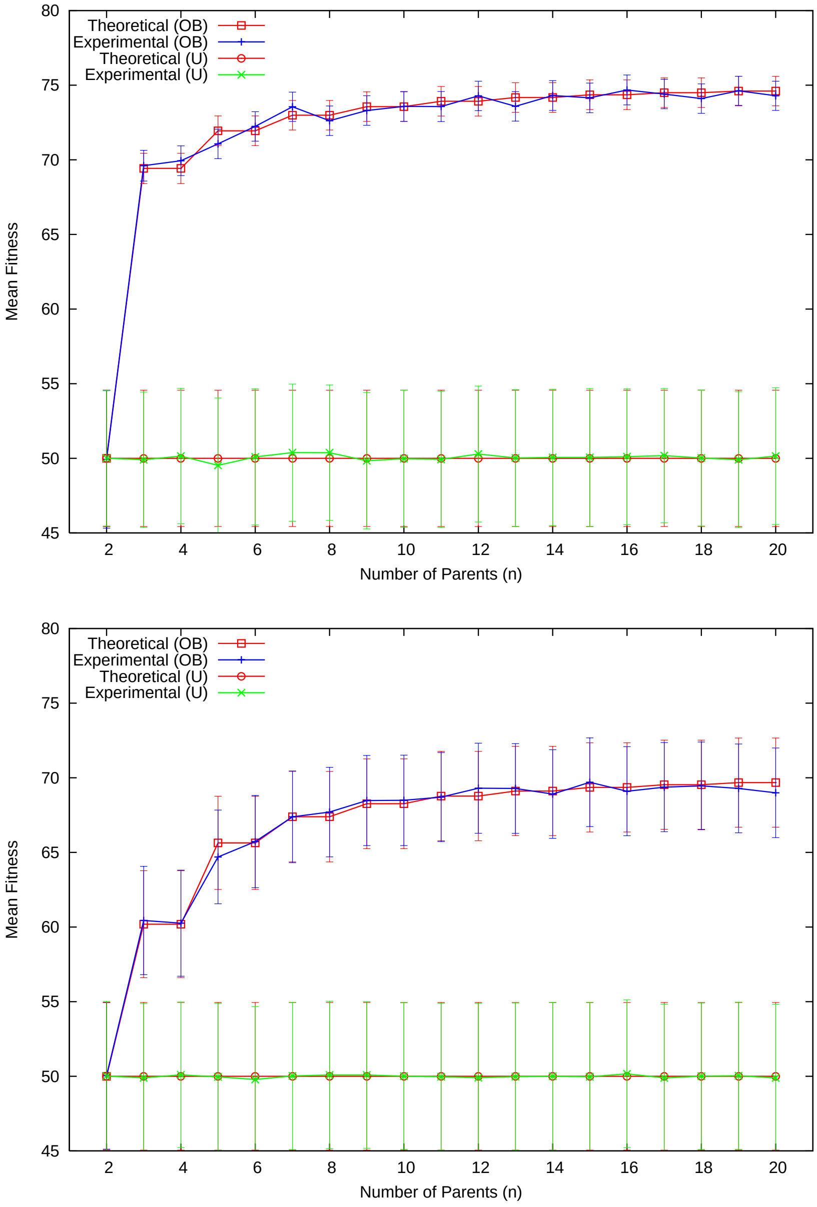
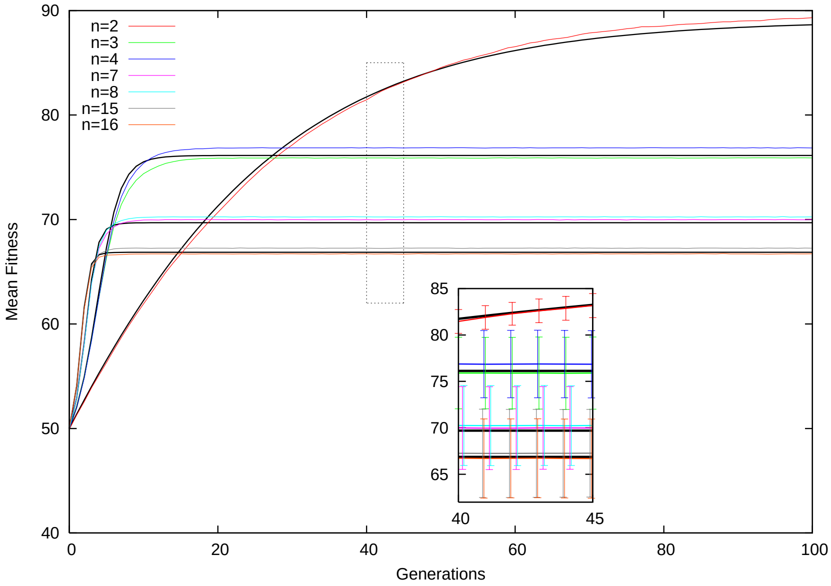


Fig. 4.17. The mean fitness for the MPGA using initialization bias $\beta = \frac{1}{32}$, random selection ($\mathcal{I} = 0$), n-parent OB-Scan and U-Scan, and bit-flip mutation with $\gamma_m = 0.01$ (top) and $\gamma_m = 0.1$ (bottom) after 1000 generations

Figures 4.18 and 4.19 compare the mean fitness computed by Theorem 4.47 with the mean fitness averaged over 100 runs. First, these figures show the theoretical and the experimental results fit very well. Second, we can see the profiles of n-parent OB-Scan with an even n are very close to those of $(n-1)$-parent OB-Scan. This confirms the *pairwise equivalence* claimed in Corollary 4.9. Third, for mutation rates $\gamma_m \geq 0.05$, OB-Scan with $n > 2$ achieves higher fitness and even faster convergence than OB-Scan with $n = 2$, viz uniform crossover. Nevertheless, there is no winner in the number of parents for all these four mutation rates: the best number of parents is dependent upon the mutation rate. It is noteworthy that the profiles of 7-parent and 8-parent OB-Scan at $\gamma_m = 0.2$ continue climbing after 20 generations while others turn into steady then. This condition also occurs for 3-parent and 4-parent OB-Scan at $\gamma_m = 0.1$. Although the mutation rates in these two cases are relatively strong to the common setting $\gamma_m = \frac{1}{l}$, they drive MPGAs using OB-Scan to advance in fitness. These results indicate the important role of mutation in the performance of MPGA using OB-Scan.

Furthermore, we examine the mean fitness for 1000 generations. Figure 4.20 depicts the experimental and the theoretical fitness means at 1000 generations ($t = 1000$); additionally it plots the theoretical mean convergence fitness ($t \to \infty$) according to Theorem 4.54. First, we can see that the theoretical and the experimental results fit very well except two cases: $n = 3, 4$ for $\gamma_m = 0.1$ and $n = 9, 10$ for $\gamma_m = 0.2$. In these two cases experimental fitness values are higher than theoretical ones. We attribute this discrepancy to the violation of the assumption: $\mathrm{Var}(\sigma) \ll \mathrm{E}^2[\sigma]$. Recall that in (4.13) we approximate the impact of selection on the gene frequency by the expectation $\mathrm{E}[\mathcal{I}^{\mathrm{p}}] = \mathcal{I}/\sqrt{l}$ based on the assumption $\mathrm{Var}(\sigma) \ll \mathrm{E}^2[\sigma]$. Once this assumption does not hold, (4.11) implies that the consequent gene frequency p_k^s will be higher than the value computed by (4.13). In the OneMax problem it leads to a higher fitness, which is reflected in the discrepancy between theoretical and experimental mean fitness in those two cases. Second, as shown in Fig. 4.20, the closeness in mean fitness between n-parent and $(n-1)$-parent OB-Scan for $n \in 2\mathbb{N}_{>1}$ reconfirms the *pairwise equivalence.* Third, the GA using 2-parent OB-Scan performs best at $\gamma_m = 0.01$ but does worst at $\gamma_m \geq 0.05$ in the experimental mean fitness at $t = 1000$. In fact, the mean fitness for 2-parent OB-Scan decreases monotonically with the increase of mutation rate. Contrary, experimental results at $t = 1000$ show that putting mutation rate up may improve MPGAs using more than two parents in OB-Scan. As $t \to \infty$, the mean convergence fitness for OB-Scan, regardless of the adopted number of parents, turns to decrease with mutation rate absolutely. Figure 4.20 further points out that, in theory, MPGAs using OB-Scan with more than two parents will converge to higher fitness than MPGAs using 2-parent OB-Scan. In addition, there exists a gap in fitness between $t = 1000$ and $t \to \infty$. Intuitively, this gap should be caused by the shortage of running generations for the test MPGAs to convergence. An advanced investigation on this gap will be conducted in Section 4.4.5.2.

Figure 4.21 compares the mean fitness of the MPGA using linear ranking selection with $\eta^+ = 1.1$ and 1.9. This figure demonstrates the impact of selection intensity on MPGAs using linear ranking selection, where the selection intensity is controlled by the



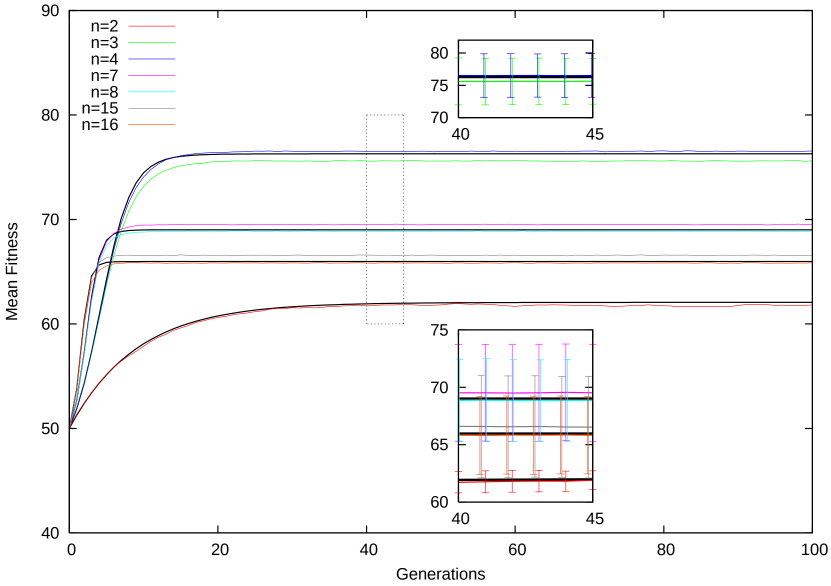
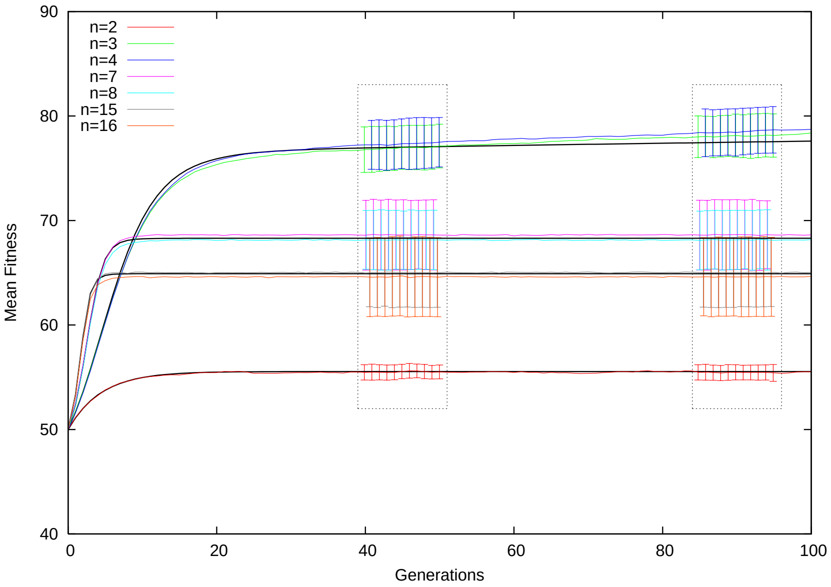


Fig. 4.18. Comparison of the mean fitness obtained from theory (black bold lines) and experiment (color thin lines) for the GA using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$ (top) and $\gamma_m = 0.05$ (bottom)



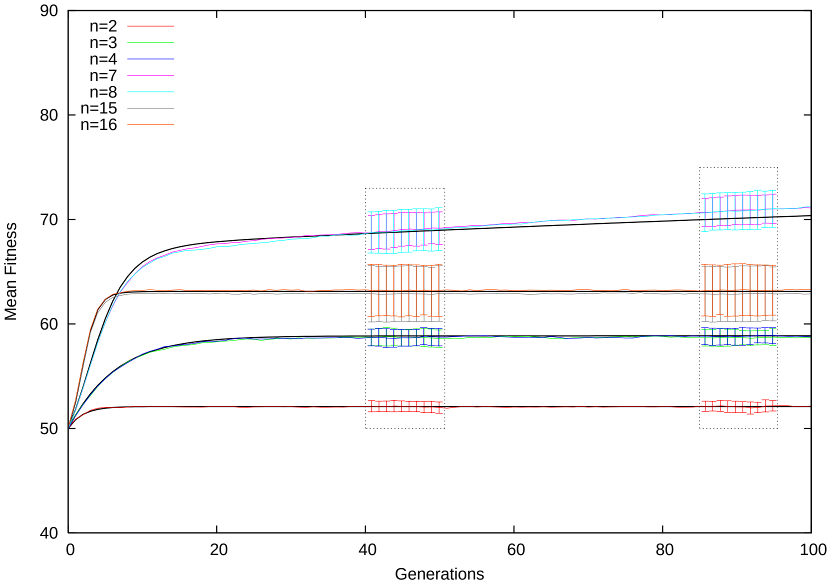
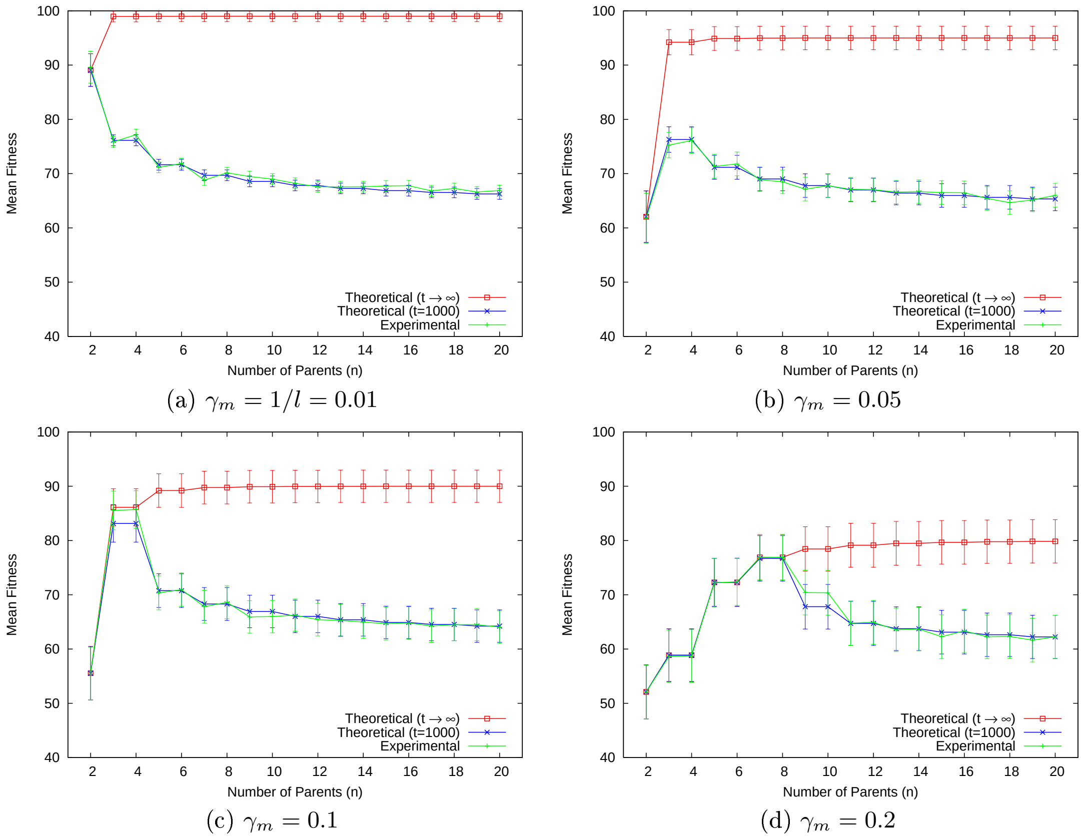


Fig. 4.19. Comparison of mean fitness obtained from theory (black bold lines) and experiment (color thin lines) for the GA using linear ranking selection with $\eta^{+} = 1.5$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.10$ (top) and $\gamma_m = 0.20$ (bottom)



(a) $\gamma_m = 1/l = 0.01$

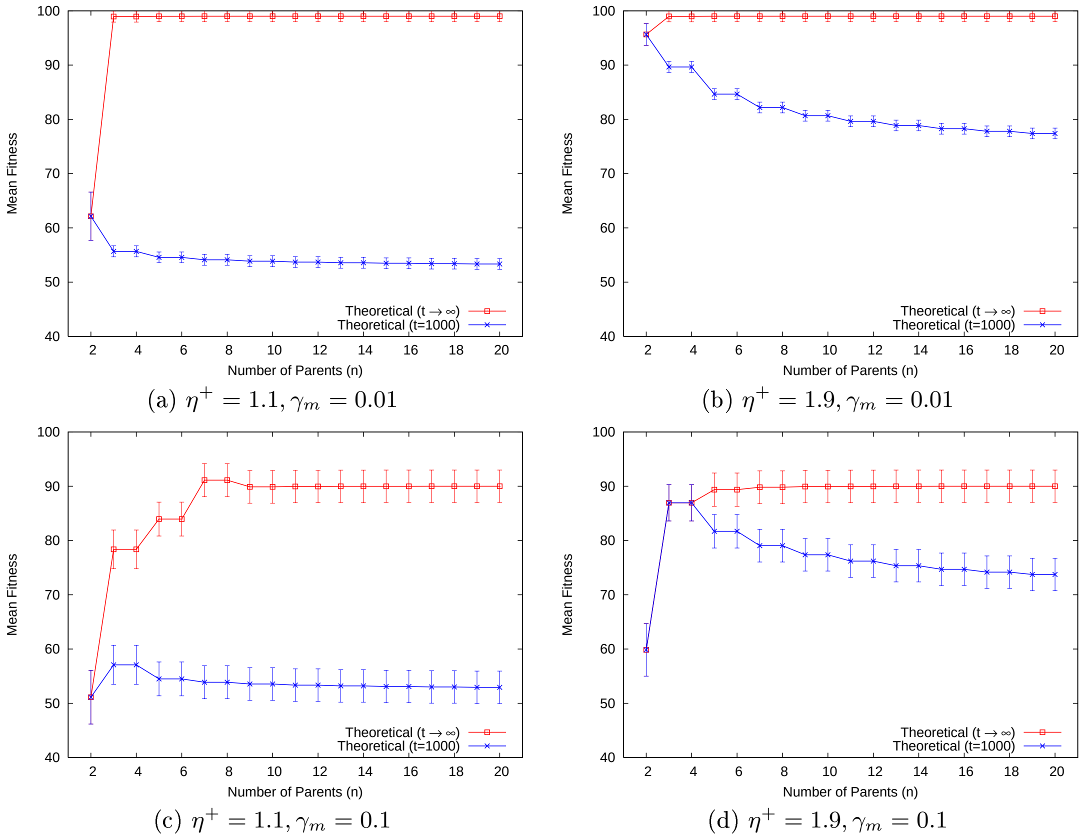
(b) $\gamma_m = 0.05$

(c) $\gamma_m = 0.1$

(d) $\gamma_m = 0.2$

Fig. 4.20. The mean fitness of the GA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$, 0.05, 0.1, and 0.2

parameter η^+. For running time $t = 1000$, the increase of η^+ from 1.1 to 1.9 significantly improves the mean fitness for any number of parents. In addition, the number of parents in OB-Scan makes little difference in mean fitness as $\eta^+ = 1.1$, which provides slight selection intensity. Precisely, the selection intensity of linear ranking selection with $\eta^+ = 1.1$ is only $\frac{1}{5}$ of that with $\eta^+ = 1.5$ and $\frac{1}{9}$ of that with $\eta = 1.9$. With such small selection intensity, MPGAs lack the driving power toward high fitness and then result in mean fitness around the initial mean fitness 50. In Section 4.1.2 we mentioned that using more than two parents in OB-Scan will intensify the tendency toward allele 0 or 1. Thus, even with slight selection intensity, MPGAs using more than two parents in OB-Scan are expected to achieve a higher mean fitness in the long term. This merit is demonstrated in the mean fitness as $t \to \infty$ in Fig. 4.21. In the next section we will further check the cost of this merit in terms of running time.



(a) $\eta^+ = 1.1, \gamma_m = 0.01$

(b) $\eta^+ = 1.9, \gamma_m = 0.01$

(c) $\eta^+ = 1.1, \gamma_m = 0.1$

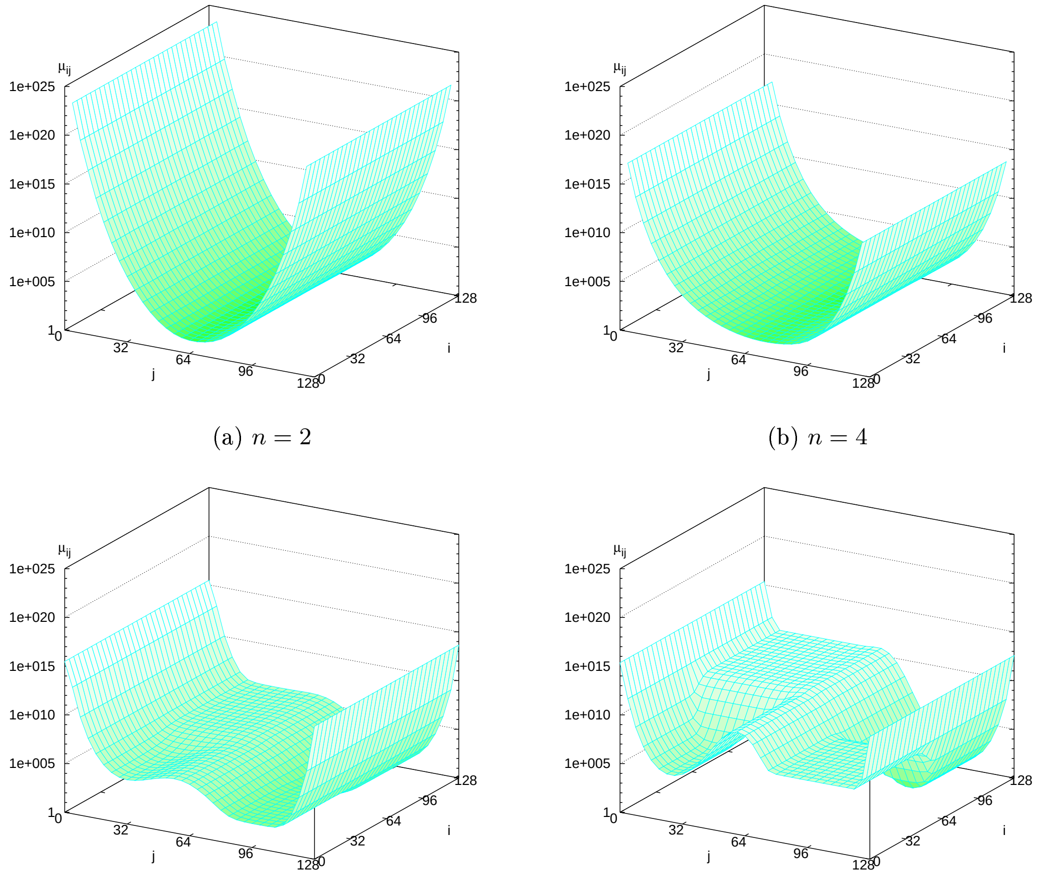
(d) $\eta^+ = 1.9, \gamma_m = 0.1$

Fig. 4.21. Comparison of the mean fitness for linear ranking selection with $\eta^+ = 1.1$ and 1.9 in the GA using n-parent OB-Scan and bit-flip mutation with $\gamma_m = 0.01$ and 0.1

4.4.5.2 Impact of Running Time

The results in the previous section showed a gap in mean fitness between running time $t = 1000$ and $t \to \infty$. A reason for this gap is that the running time $t = 1000$ is not long enough for the test MPGAs to converge. To validate this conjecture, we dive into the required time for convergence.

Figure 4.22 plots the matrix $\mathbf{M}$ of mean time from state i to state j according to the formula in Theorem 4.56. As these figures show, the contour becomes flatter but more rugged as the number of parents increases. On the one hand this flattening suggests that raising parents in OB-Scan will reduce the mean time to attain the marginal states, e.g. 0 and m. On the other hand, the ruggedness demonstrates that the mean time μ_{ij} differs for different starting states i as the number of parents is increased. That is to say, μ_{ij} becomes more dependent on state i. We further take the stationary distribution into consideration. Figure 4.23 compares the mean time μ_{ij} with the stationary distribution $\boldsymbol{\pi}$. Since the profiles of mean time μ_{ij} differ with the starting state i, this figure only shows the mean time starting from the central state, i.e.



(a) $n = 2$

(b) $n = 4$

(c) $n = 8$

(d) $n = 16$

Fig. 4.22. The mean time μ_{ij} to state j from state i for the Markov chain corresponding to the GA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan with $n = 2, 4, 8, 16$, and bit-flip mutation with mutation rate $\gamma_m = 0.2$

$i = \frac{m}{2}$. Note that Y-axis is in logarithmic scale. This figure shows that raising parents moves the valley of profiles right. This outcome reflects the proneness to higher j as the parents in OB-Scan is driven up. The stationary distribution π_j shown in the bottom half of Fig. 4.23 further validates this proneness: The distribution of profiles shifts right as the number of parents increases. It is deserving of note that the valley of profile is lifted at $n = 16$, which implies it takes longer time to attain any state on average.

The mean convergence time for mutation rate $\gamma_m = 0.2$ is presented in Fig. 4.24. It shows that the mean convergence time increases drastically for OB-Scan using more than eight parents. Moreover, this figure exposes the predefined 1000 generations for termination are insufficient for $n > 8$. As aforementioned, this insufficiency results in the fitness gap between $t = 1000$ and $t \to \infty$ in Fig 4.20. To verify this point, we

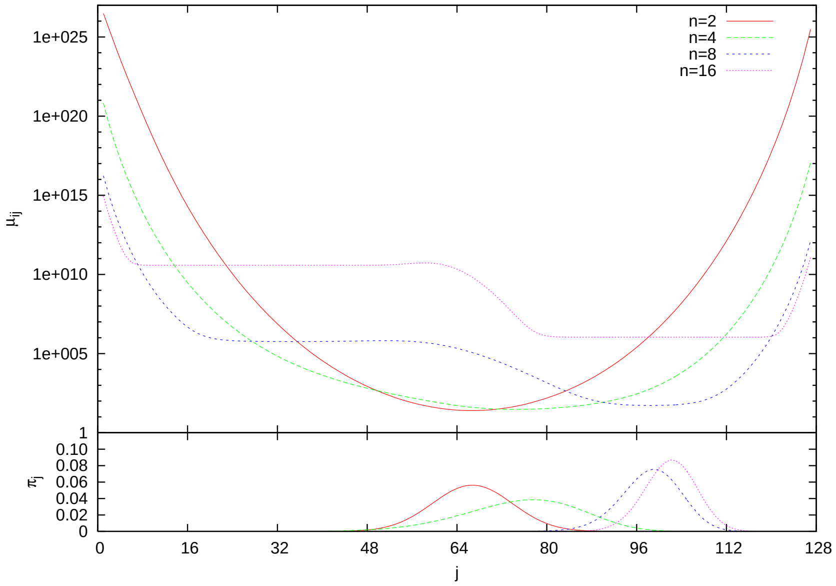


Fig. 4.23. The stationary distribution π_j and mean time μ_{ij} for $i = \frac{m}{2}$

conduct another experiment with terminal condition of 20000 generations. Figure 4.25 compares the theoretical mean convergence fitness with the experimental mean fitness obtained from 1000-generation and 20000-generation runs. According to the mean convergence time shown in Fig. 4.24, MPGAs using fewer than 11 parents are expected to converge in 20000 generations. This prediction is validated in Fig. 4.25: For OB-Scan with fewer than 11 parents, the profiles of 20000-generation experiments match those of theoretical results. This outcome reveals that using more parents in OB-Scan together with mutation can achieve better fitness than using two parents, if the running time is long enough. In other words, MPGAs using more than two parents in OB-Scan achieves better solution quality at the cost of convergence time. Nonetheless, Fig. 4.20 indicates that MPGAs using OB-Scan with $n > 2$ are capable of higher fitness in 1000 generations as mutation rate $\gamma_m > 0$. The results in Figs. 4.18 and 4.19 also show that MPGAs using OB-Scan can achieve higher fitness and even faster convergence for some numbers of parents. These outcomes validate the superiority of MPGAs using n-parent OB-Scan with $n > 2$ over GAs using 2-parent OB-Scan. That is, multi-parent OB-Scan outperforms uniform crossover.

4.4.5.3 Impact of Population Size

Here we examine the impact of population size on the performance of MPGAs. Figure 4.26 depicts the mean fitness of MPGAs for four sizes of population as $t = 1000$ and $t \to \infty$. For a limited running time $t = 1000$, the mean fitness increases with the size of population in general. On the other hand, for an infinite running time $t \to \infty$,

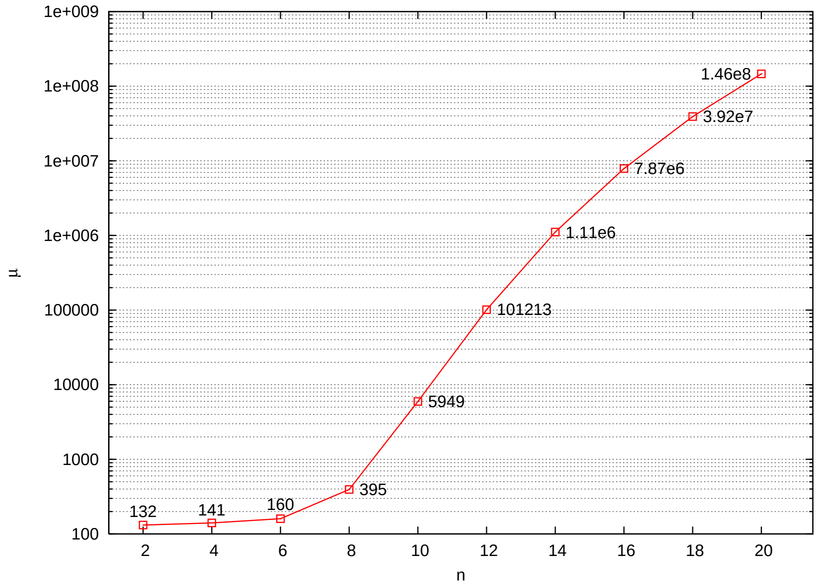


Fig. 4.24. The expected convergence time μ for the GA using linear ranking selection with $\eta^{+} = 1.50$, n-parent OB-Scan, and bit-flip mutation with mutation rate $\gamma_m = 0.2$

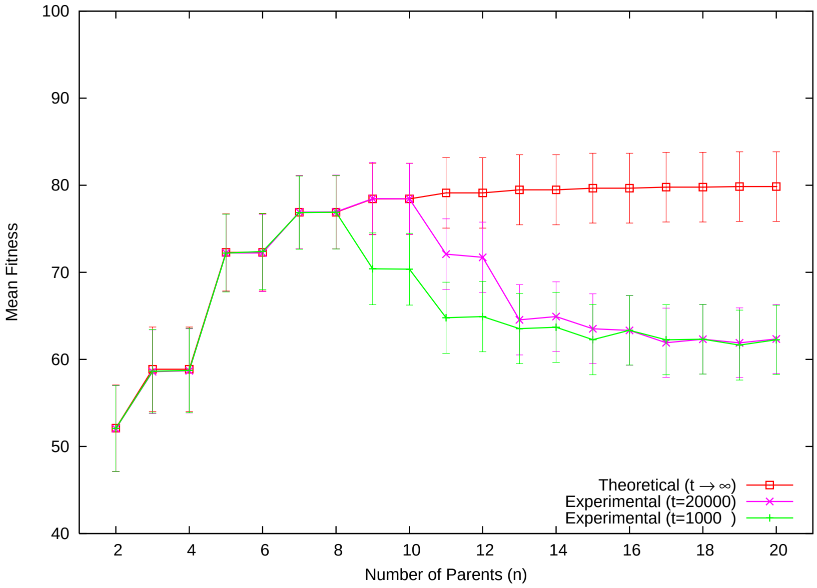
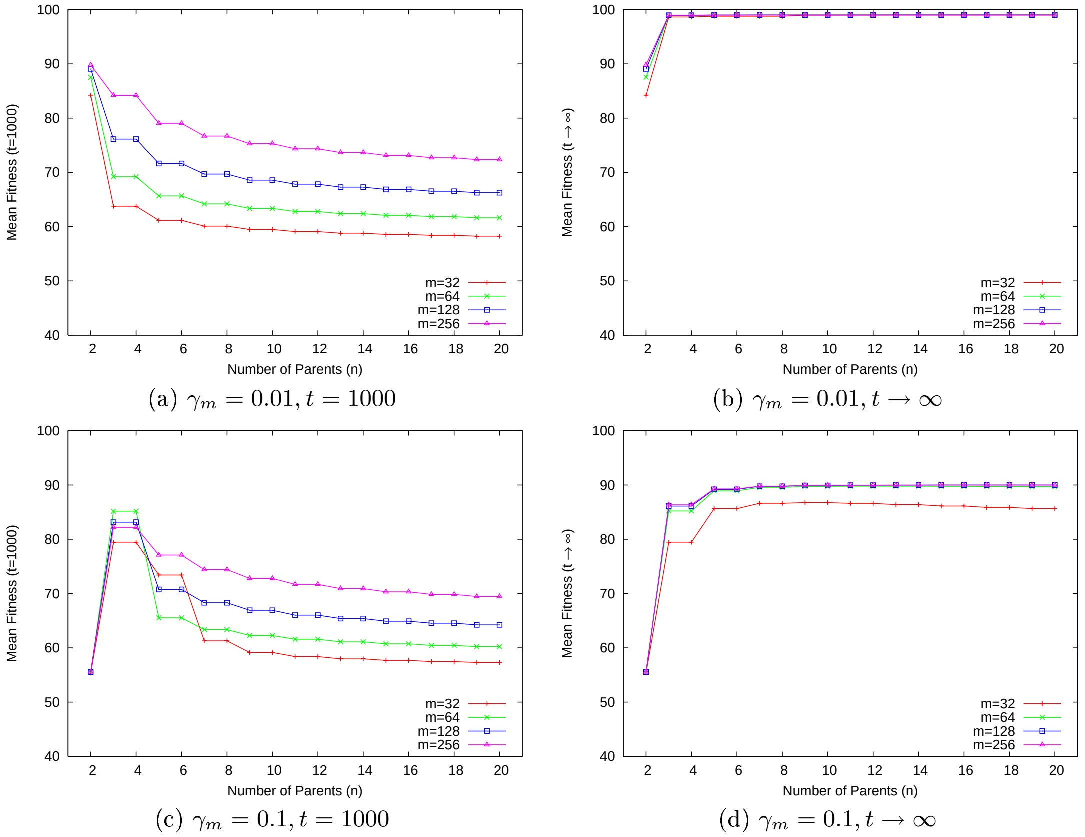


Fig. 4.25. The mean fitness of the GA using linear ranking selection with $\eta^{+} = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.2$



(a) $\gamma_m = 0.01, t = 1000$

(b) $\gamma_m = 0.01, t \rightarrow \infty$

(c) $\gamma_m = 0.1, t = 1000$

(d) $\gamma_m = 0.1, t \rightarrow \infty$

Fig. 4.26. The mean fitness for population size m = 32, 64, 128, and 256 of the GA using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$ and 0.1

the mean fitness differs slightly with respect to four population sizes. In addition, these figures exhibit the impact of running time on the mean fitness: MPGAs using more than two parents in OB-Scan gain better mean fitness as the running time increases from 1000 to an infinite number.

We look further into the impact of population size on the mean convergence time of MPGAs. Figure 4.27 demonstrates that the mean convergence time increases substantially with the size of population and this increase is reinforced by raising the number of parents. In other words, MPGAs with a larger population need more time to converge, especially when using a large number of parents in OB-Scan. As for solution quality, the increase of population size can improve the mean convergence fitness for $n = 2$ with $\gamma_m = 0.01$ (see Fig. 4.26(b)) and for $n > 2$ with $\gamma_m = 0.1$ (see Fig. 4.26(d)). The longer convergence time and the potentially better fitness caused by the increase of population size validate a conventional wisdom in the GA community: a large population causes long convergence time but may enhance the solution quality.

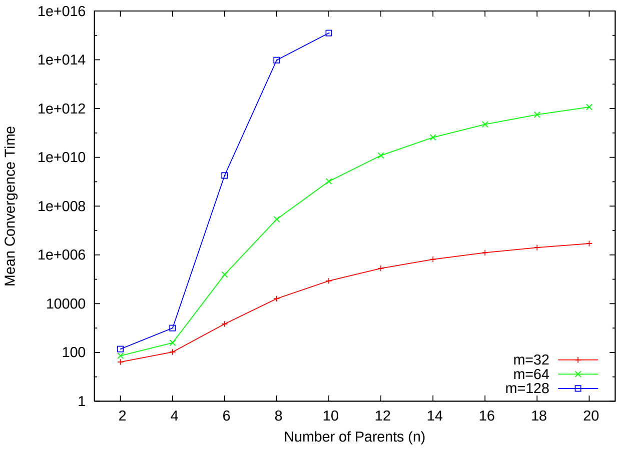


Fig. 4.27. The mean convergence time for population size $m = 32$, 64, and 128 of the GA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.1$

In the light of quick convergence, it seems ideal to adopt small population to obtain preferable convergence fitness for the use of more than two parents. However, the effect of mutation rate on the convergence time should be further considered. Figure 4.28 depicts the mean convergence for population size $m = 32$ with respect to four mutation rates. Obviously, the lower the mutation rate, the longer the mean convergence time. The use of more than two parents with a small mutation rate needs a vast number of generations to converge, even in a small population. For example, the MPGA using 4-parent OB-Scan with $\gamma_m = 0.01$ in Fig. 4.28 amounts to more than 10^7 generations. Hence, using more than two parents in OB-Scan has an inevitably long mean convergence time, except for small population with a large mutation rate, e.g. $m = 32$ with $\gamma_m = 0.2$. It is also deserving of note that neither population size nor mutation rate has a significant impact on the mean convergence time of MPGAs using 2-parent OB-Scan.

4.4.5.4 Impact of Generation Gap

This section checks the impact of generation gap on the convergence of MPGAs. Here MPGAs *without* selection are examined in order to eliminate the effect of approximation of selection intensity, as mentioned in Section 4.4.5.1.

Figure 4.29 plots the convergence of MPGAs with initialization bias $\beta = \frac{1}{32}$ for four generation gaps: $\frac{1}{128}$ (steady-state GA), $\frac{32}{128}$, $\frac{64}{128}$, and 1 (generational GA). Clearly, the theoretical results are consistent with the experimental results. This consistence validates the correctness of the theoretical model. In addition, this figure shows that these four generation gaps make no significant difference in mean fitness. In other words, the generation gap of the survivor defined in Definition 4.42 has a little impact on the performance of MPGAs. An interesting point is that MPGAs using 2-parent OB-Scan, i.e. GAs using uniform crossover, converge to a mean fitness worse than the initial

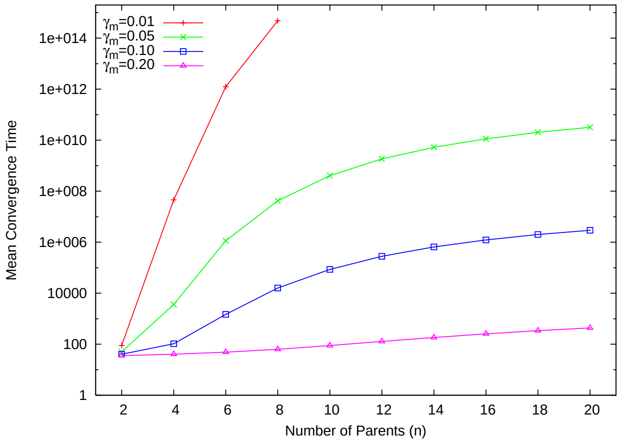
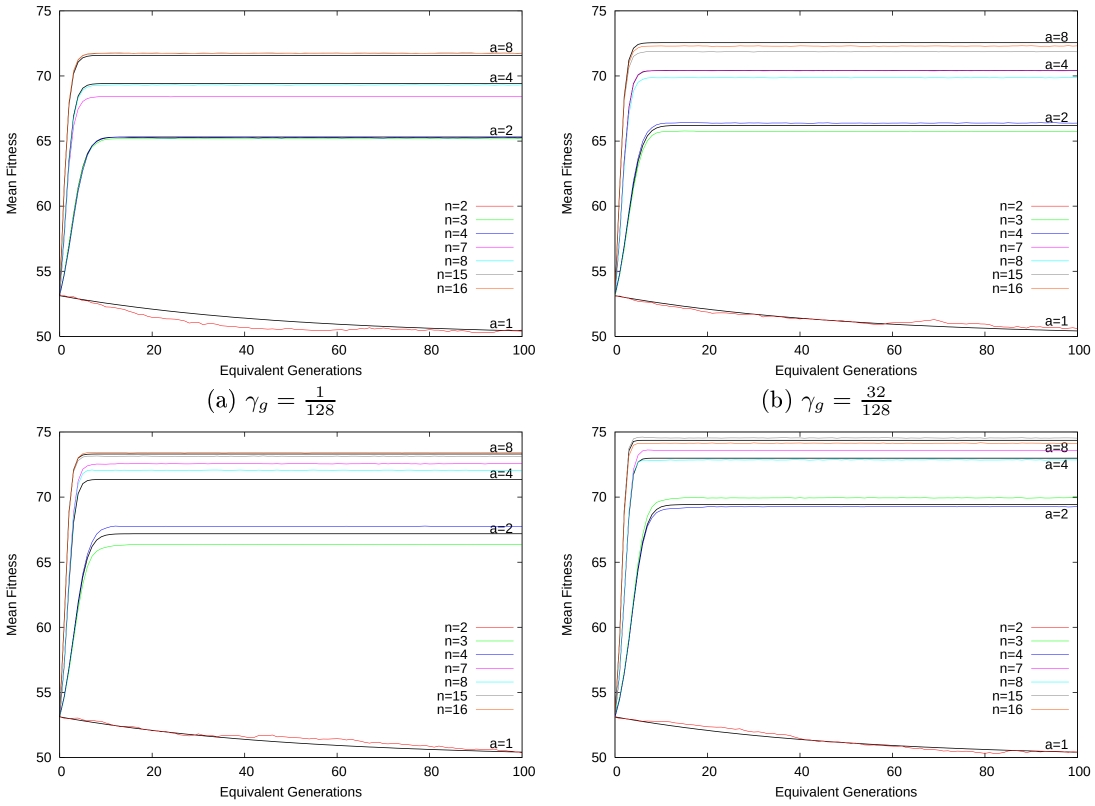


Fig. 4.28. The mean convergence time for population size $m = 32$ of the GA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$, 0.05, 0.1, and 0.2

mean fitness; on the contrary, MPGAs using OB-Scan with more than two parents converge to better mean fitness. As no selection intensity is applied, the performance of MPGAs is only subject to crossover and mutation. The analysis in Section 4.1.3 demonstrated that mutation has the effect of driving the gene frequency back to the neutral frequency 0.5. Furthermore, in Section 4.1.2 we showed that 2-parent OB-Scan does not alter gene frequency while OB-Scan with more than two parents intensifies the preference of gene frequency for allele 0 or 1. Accordingly, the performance of MPGAs using OB-Scan turns into an outcome of competition between the intensification of OB-Scan and the pullback of mutation. For MPGAs using 2-parent OB-Scan, there exists no intensification; therefore, they converge to a worse mean fitness than the biased initial mean fitness. By contrast, MPGAs using OB-Scan with more than two parents achieve better mean fitness owing to the stronger influence of OB-Scan than mutation. The phenomenon—the more the parents, the higher the mean convergence fitness—also demonstrates the intensification caused by increasing the number of parents in OB-Scan.

4.5 Convergence in the Generalized OneMax Problem

The previous section has shown the effectiveness of our theoretical analysis in the OneMax problem. In this section we further extend the analysis for a more general fitness function. The definition of this problem is given as follows.



(a) $\gamma_g = \frac{1}{128}$

(b) $\gamma_g = \frac{32}{128}$

(c) $\gamma_g = \frac{64}{128}$

(d) $\gamma_g = 1$

Fig. 4.29. The convergence of mean fitness for the MPGA using initialization bias $\beta = \frac{1}{32}$, random selection ($\mathcal{I} = 0$), n-parent OB-Scan, bit-flip mutation with mutation rate $\gamma_m = 0.01$, and generation gap $\gamma_g = \frac{1}{128}, \frac{32}{128}, \frac{64}{128}$, and 1.

Definition 4.60 (The Generalized OneMax problem). *The Generalized OneMax problem is to find the binary string* $\mathbf{c} = (c_1, \ldots, c_l) \in \{0,1\}^l$ *that maximizes the following function*

$$f(\mathbf{c}) = \sum_{k=1}^{l} \omega_k c_k,$$

where the coefficient $\omega_k \in \mathbb{R}_+$.

Remark 4.61. The classic OneMax problem and the BinInt (binary integer) problem are special cases of the Generalized OneMax problem:

1. The OneMax problem: $\omega_1 = \omega_2 = \cdots = \omega_l = 1$.
2. The BinInt problem: $\omega_k = 2^{k-1}$ for $k = 1, \ldots, l$.

The mean and the variance of fitness in the Generalized OneMax problem are presented as follows.

Lemma 4.62. *In the Generalized OneMax problem, we have the mean fitness*

$$\bar{f} = \sum_{k=1}^{l} \omega_k p_k. \tag{4.24}$$

Assume the covariance $\mathrm{Cov}(c_u, c_v) = 0$ *for any two loci* $u, v \in \{1, \dots, l\}$ *and* $u \neq v$. *The variance* σ_F^2 *of fitness is*

$$\sigma_F^2 = \sum_{k=1}^{l} \omega_k^2 \sigma_k^2 = \sum_{k=1}^{l} \omega_k^2 p_k (1 - p_k). \tag{4.25}$$

Proof. For a locus k, the gene frequency p_k represents $\mathrm{E}[c_k]$. The mean fitness can then be derived by

$$\bar{f} = \mathrm{E}[f] = \mathrm{E}\left[\sum_{k=1}^{l} \omega_k c_k\right] = \sum_{k=1}^{l} \omega_k \mathrm{E}[c_k] = \sum_{k=1}^{l} \omega_k p_k.$$

For the variance of fitness we have

$$\sigma_F^2 = \mathrm{Var}(f) = \mathrm{Var}\left(\sum_{k=1}^{l} \omega_k c_k\right) = \sum_{k=1}^{l} \omega_k^2 \mathrm{Var}(c_k) + \sum_{u \neq v} \omega_u \omega_v \mathrm{Cov}(c_u, c_v).$$

Since the covariance $\mathrm{Cov}(c_u, c_v)$ is assumed to be zero and the variance σ_k^2 stands for $\mathrm{Var}(c_k)$, we obtain

$$\sigma_F^2 = \sum_{k=1}^{l} \omega_k^2 \mathrm{Var}(c_k) = \sum_{k=1}^{l} \omega_k^2 \sigma_k^2 = \sum_{k=1}^{l} \omega_k^2 p_k (1 - p_k).$$

□

Remark 4.63. Lemma 4.33 indicates that the covariance $\mathrm{Cov}(c_u, c_v) = 0$ holds in the OneMax problem. However, this zero covariance does not necessarily hold in other Generalized OneMax problems.

In GAs, the selection operator is generally dependent upon the fitness of chromosomes. The fitness function, hence, has a direct impact on the mean fitness as well as the gene frequency affected by selection. Before investigating the variation of gene frequency, we look into the variation of mean fitness caused by selection. Recall in Section 4.4.2 the influence of selection on mean fitness is given by

$$\mathcal{I} = \frac{\bar{f}^s - \bar{f}}{\sigma_F}.$$

Let $\Delta \bar{f} = \bar{f}^s - \bar{f}$. The previous equation can be rewritten as

$$\mathcal{I} = \left(\frac{1}{\sigma_F} \right) \Delta \bar{f}.$$

This equation is analogous to the Hooke's law $F = kx$ in physics, where the selection intensity $\mathcal{I}$ is regarded as the applied force F, the fraction $\frac{1}{\sigma_F}$ is regarded as the spring constant k, and the selection differential $\Delta \bar{f}$ is regarded as the extension of spring x. Assume the selection intensity $\mathcal{I}$ has an identical effect on each locus; the variation of fitness caused by selection can be represented by a series-connected spring system, as shown in Fig. 4.30. Let x be the total extension of spring, x_i be the extension of the i^{th} spring, k be the equivalent spring constant, and k_i be the spring constant of the i^{th} spring. The spring system in Fig. 4.30 has

$$x = \frac{F}{k} = x_1 + x_2 + x_3 + x_4 = \frac{F}{k_1} + \frac{F}{k_2} + \frac{F}{k_3} + \frac{F}{k_4},$$

and, analogously, the population has

$$\Delta \bar{f} = \mathcal{I} \sigma_F = \Delta \bar{f}_1 + \Delta \bar{f}_2 + \Delta \bar{f}_3 + \Delta \bar{f}_4,$$

where $\Delta \bar{f}_k$ represents the selection differential at locus k. Motivated by this physical system, we formulate the variation of gene frequency in the following lemma.

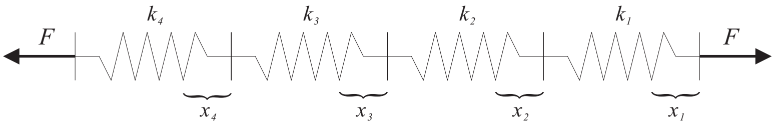


Fig. 4.30. A series connection of four springs

Lemma 4.64. *Suppose we have the gene frequency p_k of the population and the selection intensity $\mathcal{I}$. In the Generalized OneMax problem, for the gene frequency p_k^s of the parents, we have*

$$p_k^s = p_k + \mathcal{I} \left(\frac{\omega_k \sigma_k^2}{\sigma_F} \right) \tag{4.26}$$

with

$$\sigma_F = \sqrt{\sum_{k=1}^{l} \omega_k^2 \sigma_k^2} \ .$$

Proof. From Lemma 4.62 we have the variance of fitness

$$\begin{aligned}\sigma_F = \frac{\sigma_F^2}{\sigma_F} &= \frac{\omega_1^2\sigma_1^2 + \omega_2^2\sigma_2^2 + \cdots + \omega_l^2\sigma_l^2}{\sigma_F} \\ &= \left(\frac{\omega_1^2\sigma_1^2}{\sigma_F}\right) + \left(\frac{\omega_2^2\sigma_2^2}{\sigma_F}\right) + \cdots + \left(\frac{\omega_l^2\sigma_l^2}{\sigma_F}\right)\end{aligned}$$

with

$$\sigma_F^2 = \omega_1^2\sigma_1^2 + \omega_2^2\sigma_2^2 + \cdots + \omega_l^2\sigma_l^2.$$

The differential in fitness can be represented by

$$\Delta\bar{f} = \Delta\bar{f}_1 + \Delta\bar{f}_2 + \cdots + \Delta\bar{f}_l, \tag{4.27}$$

while the selection intensity gives

$$\Delta\bar{f} = \mathcal{I}\sigma_F = \mathcal{I}\left(\frac{\omega_1^2\sigma_1^2}{\sigma_F}\right) + \mathcal{I}\left(\frac{\omega_2^2\sigma_2^2}{\sigma_F}\right) + \cdots + \mathcal{I}\left(\frac{\omega_l^2\sigma_l^2}{\sigma_F}\right). \tag{4.28}$$

From Lemma 4.62, we know

$$\Delta\bar{f}_k = \bar{f}_k^s - \bar{f}_k = \omega(p_k^s - p_k).$$

Comparing (4.27) and (4.28) leads to

$$\Delta\bar{f}_k = \mathcal{I}\left(\frac{\omega_k^2\sigma_k^2}{\sigma_F}\right) = \omega(p_k^s - p_k).$$

Hence the gene frequency of the selected parents is

$$p_k^s = p_k + \mathcal{I}\left(\frac{\omega_k\sigma_k^2}{\sigma_F}\right).$$

□

Corollary 4.65. *In the OneMax problem, assuming $p_1 = p_2 = \cdots = p_l$, we have the gene frequency p_k^s of the selected parents*

$$p_k^s = p_k + \left(\frac{\mathcal{I}}{\sqrt{l}}\right)\sigma_k.$$

Proof. In the OneMax problem, the coefficients are $\omega_1 = \omega_2 = \cdots = \omega_l = 1$. In addition, the assumption $p_1 = p_2 = \cdots = p_l$ gives $\sigma_1 = \sigma_2 = \cdots = \sigma_l$. According to Lemma 4.64 and $p_1 = p_2 = \cdots = p_l$, the gene frequency is

$$\begin{aligned}p_k^s = p_k + \mathcal{I}\left(\frac{\omega_k\sigma_k^2}{\sigma_F}\right) &= p_k + \mathcal{I}\left(\frac{\sigma_k^2}{\sqrt{\sigma_1^2 + \sigma_2^2 \cdots + \sigma_l^2}}\right) \\ = p_k + \mathcal{I}\left(\frac{\sigma_k^2}{\sqrt{l\sigma_k^2}}\right) &= p_k + \left(\frac{\mathcal{I}}{\sqrt{l}}\right)\sigma_k.\end{aligned}$$

□

This corollary corresponds to (4.13); nonetheless, the assumption $p_1 = \cdots = p_l$ in Corollary 4.65 implies $\mathrm{Var}(\sigma) = 0$, which is stricter than the assumption $\mathrm{Var}(\sigma) \ll \mathrm{E}^2[\sigma]$ in (4.13). The assumption $p_1 = \cdots = p_l$ is expected to hold for the OneMax problem but not necessarily for other Generalized OneMax problems. The BinInt problem, for example, has the specific phenomenon "domino convergence" [64, 100], where the convergent sequence of genes follows the salience of coefficient ω_k in the fitness function. Therefore, it needs to consider each locus individually for the Generalized OneMax problems. The Markov model for gene frequency proposed in Definition 4.17 is based on a single locus. To take each locus into account for the Generalized OneMax problem, we expand the single-locus Markov model to the following multi-loci Markov model.

Definition 4.66 (Markov Model for Multiple Gene Frequencies). *The Markov chain $\{G(t)\}$ for gene frequencies at all loci are defined by:*

1. *The* state *is expressed by a $(m{+}1)$-nary number and the digit represents the number of allele* 1 *at the corresponding locus. The state space is thereby $\{0, 1, \ldots, M\}$ with $M = (m+1)^l - 1$. Let $i_{(k)}$ be the k^{th} digit of the $(m{+}1)$-nary number corresponding to state i. A state i gives the gene frequency at locus k:*
$$p_k = \frac{i_{(k)}}{m}.$$
2. *The transition matrix of $\{G(t)\}$ is*
$$\mathbf{P} \stackrel{\mathrm{def}}{=} \begin{pmatrix} \rho_{00} & \rho_{01} & \cdots & \rho_{0M} \\ \rho_{10} & \rho_{11} & \cdots & \vdots \\ \vdots & \vdots & & \vdots \\ \rho_{M0} & \cdots & \cdots & \rho_{MM} \end{pmatrix},$$
where ρ_{ij} denotes the transition probability of state i to state j:
$$\rho_{ij} \stackrel{\mathrm{def}}{=} \Pr\{G(t+1) = j \mid G(t) = i\}.$$

Example 4.67. The following matrix demonstrates a transition matrix for $m = 4$ and $l = 4$. The state $i = 168$ represents a pentanary number $(1133)_5$, indicating that there exist three allele 1 at locus 1 and locus 2, and one allele 1 at locus 3 and locus 4.

$$\mathbf{P} = \begin{array}{r} \\ 0 = (0000)_5 \\ 1 = (0001)_5 \\ \vdots \\ 168 = (1133)_5 \\ \vdots \\ 624 = (4444)_5 \end{array} \begin{array}{c} \begin{array}{cccc} 0 & 1 & \cdots\cdots\cdots & 624 \end{array} \\ \begin{pmatrix} 0.26 & 0.08 & \cdots\cdots\cdots & 0.03 \\ 0.16 & 0.07 & \cdots\cdots\cdots & 0.01 \\ \vdots & \vdots & & \vdots \\ 0.08 & 0.01 & & 0.15 \\ \vdots & \vdots & & \vdots \\ 0.02 & 0.01 & \cdots\cdots\cdots & 0.33 \end{pmatrix} \end{array}$$

The transition probability of the Markov chain corresponding to MPGAs is further given below.

Theorem 4.68. *For a GA using selection with selection intensity $\mathcal{I}$, n-parent OB-Scan, bit-flip mutation with mutation rate γ_m, and survivor with generation gap $\gamma_g = \frac{\lambda}{m}$, the transition probability ρ_{ij} of the Markov chain $\{G(t)\}$ corresponding to that GA in the Generalized OneMax problem is*

$$\rho_{ij} = \prod_{k=1}^{l} \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) \cdot H(j_{(k)} - x; m - \lambda, i_{(k)}, m),$$

where $i_{(k)}$ denotes the k^{th} digit of the $(m+1)$-nary number associated with i and the gene frequency

$$p'_k = I_{p^s_k}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right) + \gamma_m \left(1 - 2I_{p^s_k}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right)\right)$$

with

$$p^s_k = \frac{i_{(k)}}{m} + \mathcal{I}\left(\frac{\omega_k i_{(k)}\left(m - i_{(k)}\right)}{m\sqrt{\sum_{k=1}^{l} \omega_k^2 i_{(k)}\left(m - i_{(k)}\right)}}\right).$$

Proof. According to Definition 4.66, the number of allele 1 at locus k corresponding to state i is $i_{(k)}$. Thus the transition probability ρ_{ij} implies the gene frequency $p_k = \frac{i_{(k)}}{m}$. From Lemmas 4.7, 4.12, and 4.64, we can derive the gene frequency p'_k of the offspring reproduced by the defined GA:

$$p'_k = p^x_k + \gamma_m\left(1 - 2p^x_k\right) = I_{p^s_k}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right) + \gamma_m \left(1 - 2I_{p^s_k}\left(\left\lceil \frac{n}{2} \right\rceil, \left\lceil \frac{n}{2} \right\rceil\right)\right)$$

with

$$p^s_k = p_k + \mathcal{I}\left(\frac{\omega_k \sigma_k^2}{\sqrt{\sum_{k=1}^{l} \omega_k^2 \sigma_k^2}}\right) = \frac{i_{(k)}}{m} + \mathcal{I}\left(\frac{\omega_k i_{(k)}\left(m - i_{(k)}\right)}{m\sqrt{\sum_{k=1}^{l} \omega_k^2 i_{(k)}\left(m - i_{(k)}\right)}}\right).$$

Moreover, the proof of Theorem 4.43 gives the transition probability for locus k:

$$\Pr\{G_k(t+1) = j_{(k)} \mid G_k(t) = i_{(k)}\} = \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) \cdot H(j_{(k)} - x; m - \lambda, i_{(k)}, m).$$

Thus, the transition probability of the chain $\{G(t)\}$ can be obtained by

$$\begin{aligned}\rho_{ij} &= \prod_{k=1}^{l} \Pr\{G_k(t+1) = j_{(k)} \mid G_k(t) = i_{(k)}\} \\ &= \prod_{k=1}^{l} \sum_{x=0}^{\lambda} B(x; \lambda, p'_k) \cdot H(j_{(k)} - x; m - \lambda, i_{(k)}, m).\end{aligned}$$

□

The Markov chain $\{G(t)\}$ is a multi-loci extension of the chain $\{G_k(t)\}$. The homogeneity and ergodicity of the chain $\{G_k(t)\}$ (see Section 4.4.3) also hold for the chain $\{G(t)\}$. According to the homogeneity of $\{G(t)\}$, we can use (4.6) to compute state distribution $\boldsymbol{\pi}(t)$ at generation t.

Theorem 4.69. *Suppose we have the state distribution $\boldsymbol{\pi}(t) = (\pi_1(t), \ldots, \pi_M(t))$ corresponding to the GA defined in Theorem 4.68. In the Generalized OneMax problem, for the mean $\bar{f}(t)$ and the variance $\sigma_F^2(t)$ of fitness at generation t, we have*

$$\bar{f}(t) = \frac{1}{m}\sum_{k=1}^{l}\sum_{j=1}^{M}\omega_k j_{(k)}\pi_j(t)$$

$$\sigma_F^2(t) = \frac{1}{m^2}\sum_{k=1}^{l}\sum_{j=1}^{M}\omega_k^2 j_{(k)}(m - j_{(k)})\pi_j(t)$$

Proof. The state distribution $\boldsymbol{\pi}(t)$ gives the gene frequency at locus k:

$$p_k(t) = \sum_{j=0}^{M}\left(\frac{j_{(k)}}{m}\right)\Pr\{G(t) = j\} = \frac{1}{m}\sum_{j=0}^{M} j_{(k)}\pi_j(t).$$

Referring to Lemma 4.62, we have the mean fitness

$$\bar{f}(t) = \sum_{k=1}^{l}\omega_k p_k(t) = \frac{1}{m}\sum_{k=1}^{l}\sum_{j=1}^{M}\omega_k j_{(k)}\pi_j(t),$$

and the variance of fitness

$$\begin{aligned}\sigma_F^2(t) &= \sum_{k=1}^{l}\omega_k^2\left(p_k(t) - p_k^2(t)\right)\\ &= \sum_{k=1}^{l}\sum_{j=1}^{M}\omega_k^2\left(\frac{j_k}{m} - \left(\frac{j_k}{m}\right)^2\right)\pi_j(t)\\ &= \frac{1}{m^2}\sum_{k=1}^{l}\sum_{j=1}^{M}\omega_k^2 j_{(k)}(m - j_{(k)})\pi_j(t).\end{aligned}$$

□

It is noteworthy that the Markov chain $\{G(t)\}$ has a large transition matrix, even for a small population size and a small problem size. Precisely, the transition matrix $\mathbf{P}$ has $(m+1)^{2l}$ elements. A 100-bit Generalized OneMax problem with population size $m = 128$, for instance, has a transition matrix consisting of $129^{200} \approx 1.312 \times 10^{422}$ elements. Such a matrix is so demanding in memory space that it becomes unpractical for analysis on large problems. In the following two sections, we will make use of this Markov model to analyze the convergence of MPGAs in two Generalized OneMax problems restricted by a small length l.

4.5.1 Case 1: The Proportionate OneMax Problem

We first analyze the convergence of MPGAs using OB-Scan in the Proportionate OneMax problem, where the coefficients ω_k are proportional to the locus k.

Definition 4.70 (The Porportionate OneMax Problem). *The Proportionate OneMax problem is to find the binary string* $\mathbf{c} = (c_1, \ldots, c_l) \in \{0,1\}^l$ *that maximizes the following function* $f : \{0,1\}^l \to \{0, 1, \ldots, \frac{l(l+1)}{2}\}$*:*

$$f(\mathbf{c}) = \sum_{k=1}^{l} k \cdot c_k.$$

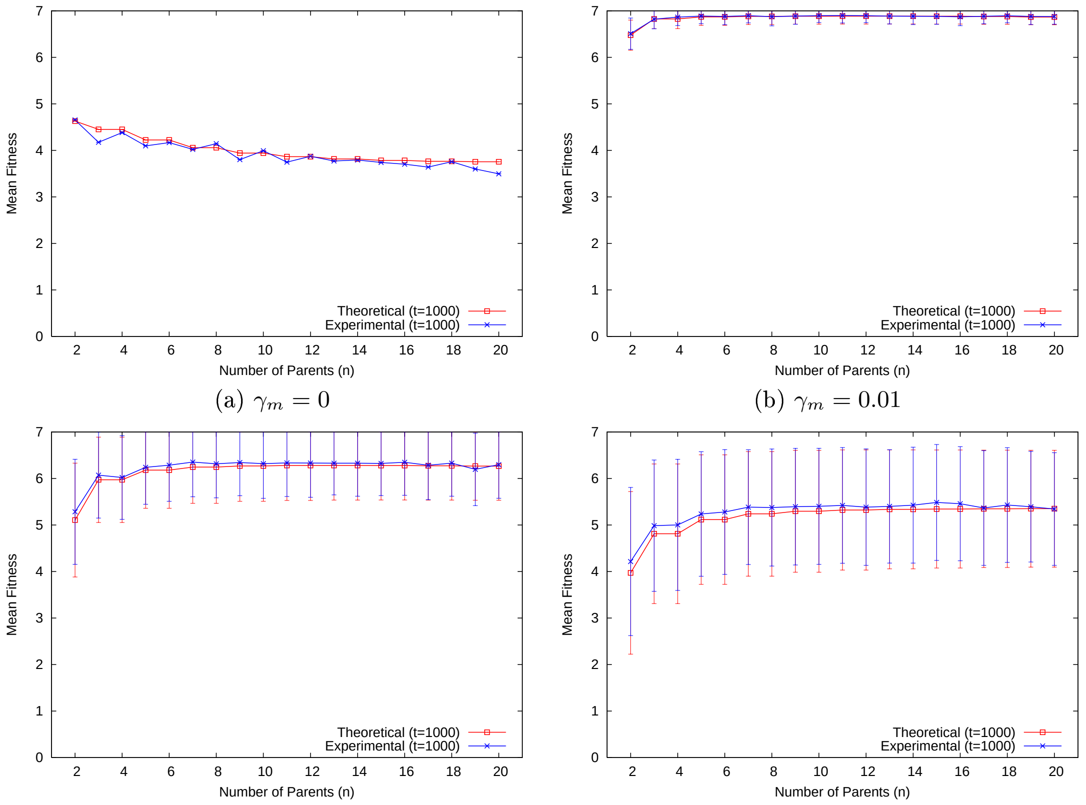
The Proportionate OneMax problem is a special case ($\omega_k = k$) of the Generalized OneMax problem. Using Theorem 4.69, we can compute the mean and the variance of fitness in the Proportionate OneMax problem:

$$\bar{f}(t) = \frac{1}{m} \sum_{k=1}^{l} \sum_{j=1}^{M} k \cdot j_{(k)} \pi_j(t) \tag{4.29}$$

$$\sigma_F^2(t) = \frac{1}{m^2} \sum_{k=1}^{l} \sum_{j=1}^{M} k^2 \cdot j_{(k)}(m - j_{(k)}) \pi_j(t). \tag{4.30}$$

As aforementioned, the size of transition matrix for multiple gene frequencies increases exponentially with the size of problem, namely the chromosome length l in the Generalized OneMax problem. Thus we only conduct theoretical analysis on the 4-bit Proportionate OneMax problem with population size $m = 4$. Following Definition 4.66, the Markov chain $\{G(t)\}$ has a state space $\{0, 1, \ldots, 5^4 - 1\}$ and its transition matrix is a $\mathbb{R}^{625 \times 625}$ square matrix. The setting of MPGAs for experiments follows that used in Section 4.4.5. Each experiment includes 1000 independent runs.

Figure 4.31 compares the mean fitness computed by (4.29) and (4.30) with the mean fitness averaged over 1000 runs. The convergence of the mean fitness is additionally depicted in Figs. C.1 and C.2. First, these figures show that the theoretical results are consistent with the experimental results. However, their discrepancy becomes more and more apparent as the mutation rate increases. We attribute the discrepancy to the violation of the assumption in Lemma 4.62: the covariance $\mathrm{Cov}(c_u, c_v) = 0$ for any two loci $u, v \in \{1, \ldots, l\}$. This violation will affect the estimation of variance σ_F^2 in (4.25) and further the computation of gene frequency p_k^s in (4.26). Second, these figures demonstrate the impact of mutation on the performance of MPGAs using OB-Scan. For $\gamma_m = 0$, increasing the number of parents receives a faster convergence but ends up with a worse solution — a typical phenomenon of premature convergence. By contrast, in the cases of $\gamma_m > 0$, using more than two parents in OB-Scan gains faster convergence and better solution quality. This superiority in both convergence speed and solution quality validates the merits of MPGAs using OB-Scan. It is noteworthy



(a) $\gamma_m = 0$

(b) $\gamma_m = 0.01$

(c) $\gamma_m = 0.05$

(d) $\gamma_m = 0.1$

Fig. 4.31. The mean and the standard deviation of fitness for the MPGA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$, 0.05, 0.1, and 0.2 at $t = 1000$ generations

that the higher the mutation rate, the lower the mean convergence fitness. In addition, there is no winner in the number of parents for all mutation rates. The optimal number of parents for MPGAs using OB-Scan is dependent upon mutation rate.

4.5.2 Case 2: The BinInt Problem

This section further investigates the convergence of MPGAs using OB-Scan in the BinInt (binary integer) problem. The definition of the BinInt problem is given below.

Definition 4.71 (The BinInt Problem). *The BinInt problem is to find the binary string* $\mathbf{c} = (c_1, \ldots, c_l) \in \{0,1\}^l$ *that maximizes the following function* $f : \{0,1\}^l \to \{0, 1, \ldots, 2^l - 1\}$*:*

$$f(\mathbf{c}) = \sum_{k=1}^{l} 2^{k-1} c_k.$$

The BinInt is also a special case ($\omega_k = 2^{k-1}$) of the Generalized OneMax problem. Therefore, we can use Theorem 4.69 to compute the mean and the variance of fitness in the BinInt problem:

$$\bar{f}(t) = \frac{1}{m}\sum_{k=1}^{l}\sum_{j=1}^{M} 2^{k-1} j_{(k)} \pi_j(t) \tag{4.31}$$

$$\sigma_F^2(t) = \frac{1}{m^2}\sum_{k=1}^{l}\sum_{j=1}^{M} 2^{2(k-1)} j_{(k)}(m - j_{(k)}) \pi_j(t). \tag{4.32}$$

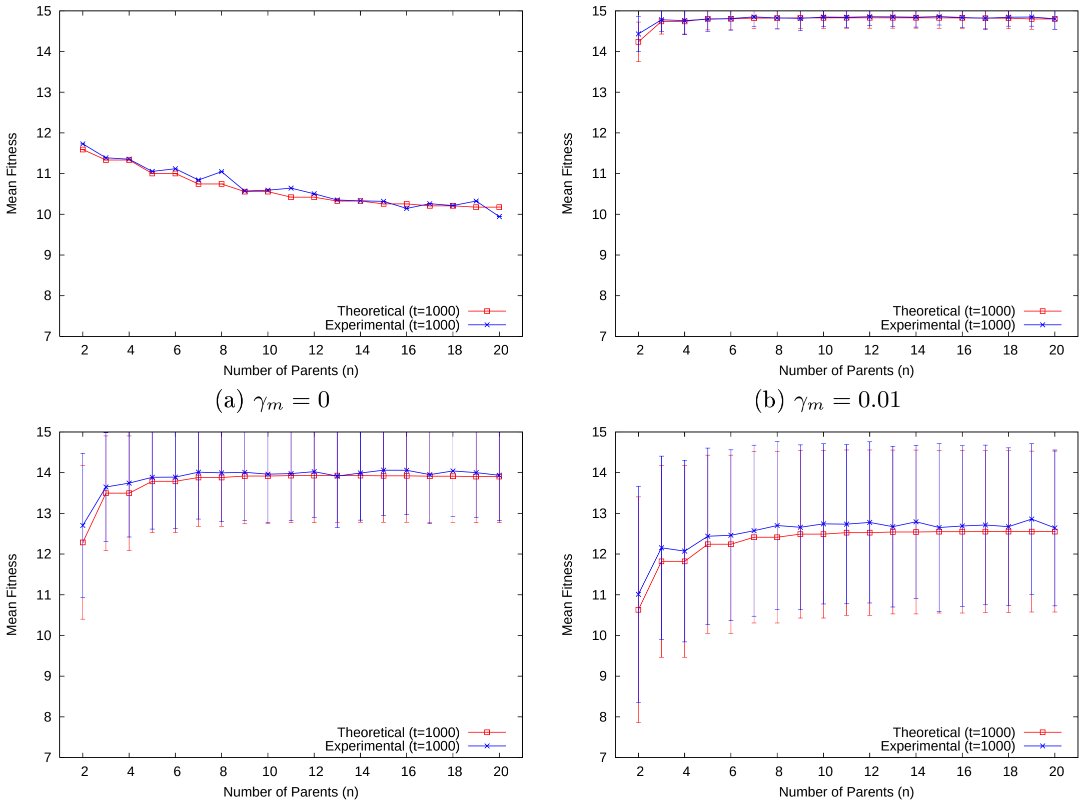
Subject to memory space, here we only analyze the 4-bit BinInt problem with population size $m = 4$. As indicated in the previous section, the corresponding Markov chain $\{G(t)\}$ has a transition matrix with 625×625 elements. For experimental validation, the setting of MPGAs follows that used in Section 4.4.5. Each experiment includes 1000 independent runs.

Figure 4.32 compares the theoretical mean fitness and the experimental one at $t = 1000$ for four different mutation rates. The convergence of the mean fitness is plotted in Figs. C.3 and C.4. In general, the theoretical results and the experimental ones fit well. Similar to the results in the Proportionate OneMax problem, the discrepancy between the theoretical results and the experimental results becomes apparent as the mutation rate increases. Moreover, mutation has similar influences on the convergence of MPGAs using OB-Scan: Increasing the number of parents incurs premature convergence and harms the mean fitness for MPGAs without mutation. Nevertheless, as mutation is adopted, using more than two parents turns into be beneficial. For the mutation rate $\gamma_m > 0$, MPGAs using n-parent OB-Scan with $n > 2$ all outperform those using 2-parent OB-Scan in terms of convergence speed and solution quality. These preferable outcomes in convergence speed and solution quality substantially reconfirm the benefits of using more than two parents in MPGAs using OB-Scan. Additionally, the optimal number of parents in OB-Scan hinges upon the mutation rate.

4.6 Beyond the OneMax Problem Domain

In this section we conduct experiments to examine the applicability of our theoretical arguments to other problem domains. The test suite includes the extended De Jong's second function (F2), the Rastrigin (RAS), the Schwefel (SCH), and the Griewangk (GRI) functions[4]. The properties and the parameters of these functions follow those listed in Table 3.1. Note that these functions are minimization problems: the lower the fitness, the better the performance. The setting of MPGAs is presented in Table 4.1. Owing to the long running time (10^4 generations), each experiment includes only 10 independent runs.

[4] A more detailed description of these functions is given in Appendix A.



(a) $\gamma_m = 0$

(b) $\gamma_m = 0.01$

(c) $\gamma_m = 0.05$

(d) $\gamma_m = 0.1$

Fig. 4.32. The mean and the standard deviation of fitness for the MPGA using linear ranking selection with $\eta^+ = 1.50$, n-parent OB-Scan, and bit-flip mutation with $\gamma_m = 0.01$, 0.05, 0.1, and 0.2 at $t = 1000$ generations

Table 4.1. The setting of MPGAs in experiments

GA type	Generational GA
Representation	Bit string
Population size	32
Selection	Linear ranking selection with bias 1.5
Crossover	OB-Scan with $n = 2, \ldots, 20$
Crossover rate	1.0
Mutation	Bit-flip mutation
Mutation rate	0.01, 0.05, 0.1, and 0.2
Termination	10^4 generations

Figure 4.33 depicts the mean of the best fitness at $t = 10^4$ on the four test functions. The progress of convergence is additionally plotted in Figs. C.5 and C.6. These experimental results agree well with our theoretical arguments about MPGAs in the OneMax problem domain: MPGAs using OB-Scan with $n > 2$ gain better fitness than GAs ($n = 2$) as mutation rate $\gamma_m > 0.01$ on these four test functions. The best number of parents, nevertheless, depends upon the adopted mutation rate: uniform crossover ($n = 2$) performs best for $\gamma_m = 0.01$; OB-Scan with $n = 3$ (or 4) performs best for $\gamma_m = 0.05$ and 0.1; OB-Scan with $n = 9$ (or 10) performs best for $\gamma_m = 0.2$. These outcomes are analogous with those in the OneMax problem, as shown in Fig. 4.20. Moreover, the experimental results in Fig. 4.33 confirm that the mutation rate should be properly increased for better fitness, when raising the number of parents in OB-Scan. In addition, the phenomenon of pairwise equivalence occurs partly in Fig. 4.33.

Figure 4.33 shows that GAs with $\gamma_m = 0.01$ achieve the best fitness on all the four test functions; that is, GAs outperform MPGAs. However, as discussed in Section 4.4.5.2, an insufficient running time will depress the performance of MPGAs, since they need longer convergence time when using more than two parents in OB-Scan. We therefore conduct additional experiments with 10^6 generations as the termination crite-

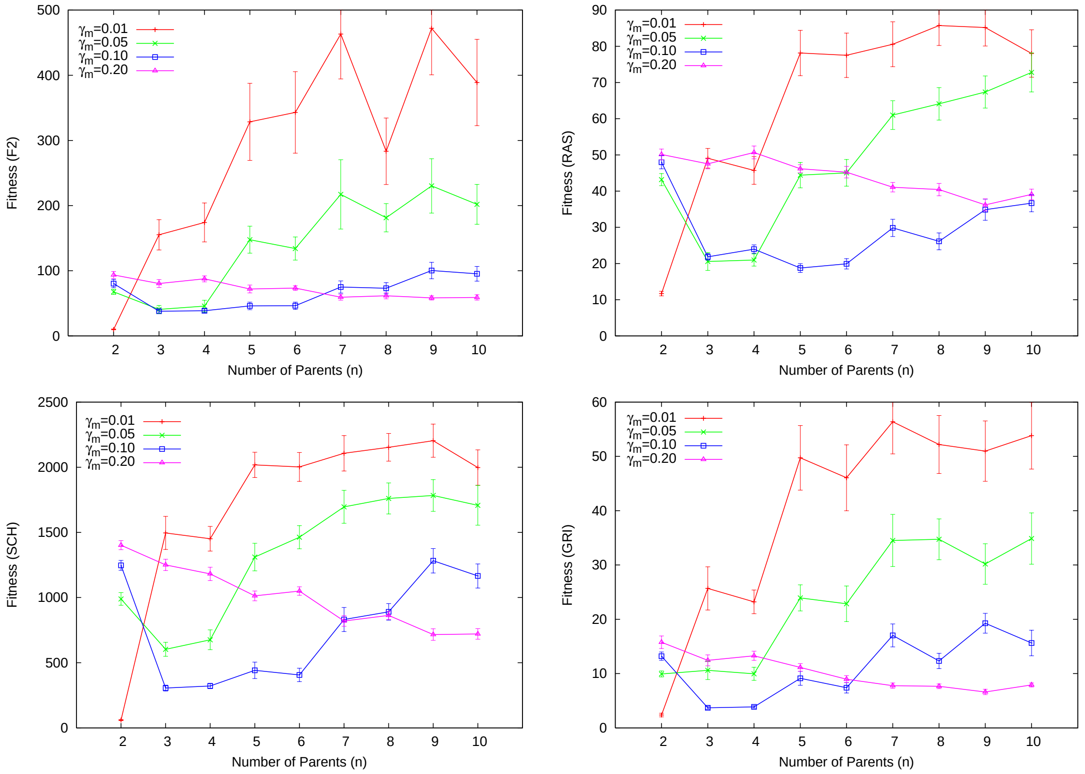
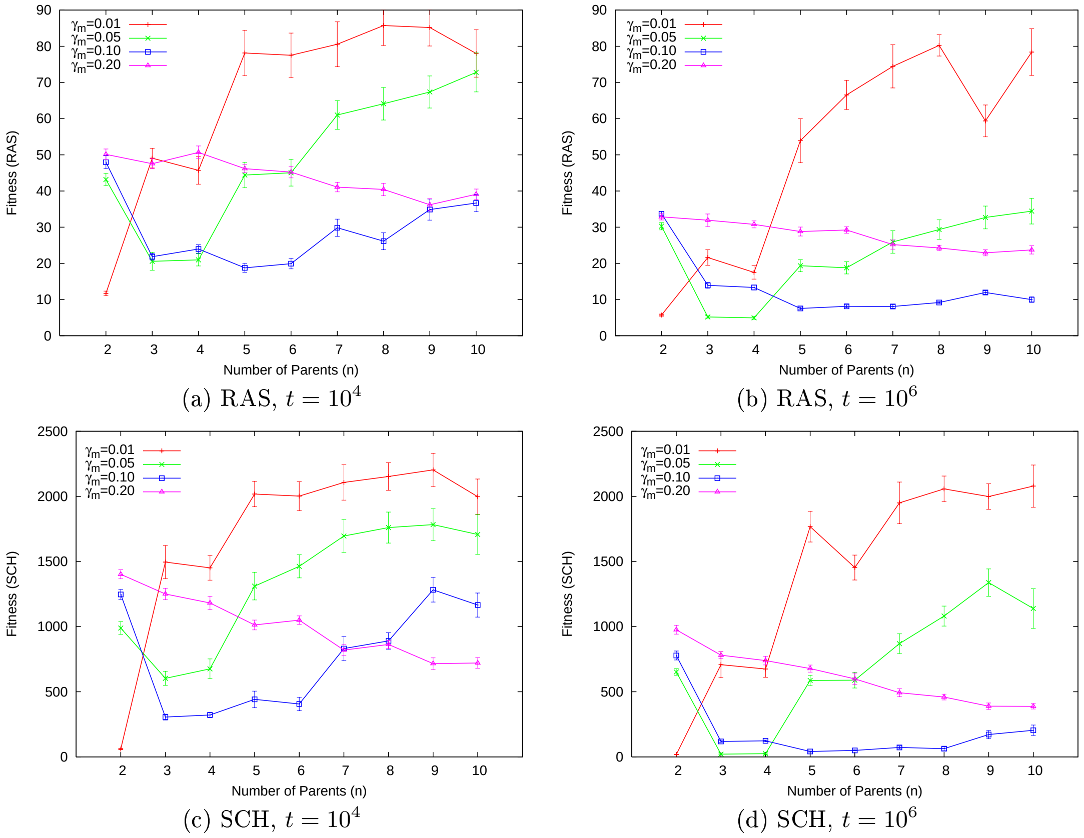


Fig. 4.33. The mean and the standard error of the best fitness at $t = 10^4$ on the F2, RAS, SCH, and GRI functions over 10 runs

rion. Figure 4.34 compares the mean of the best fitness of 10^4-generation runs and that of 10^6-generation runs on the RAS and the SCH functions. The convergence of MPGAs on these two functions in 10^6 generations is depicted in Fig. C.7. These experimental results demonstrate that the prolongation of running time substantially improves the best fitness of MPGAs as $n > 2$. They also indicate that MPGAs using OB-Scan with $n = 3$ (or 4) can outperform GAs, albeit slightly. The favorable setting, $n = 3$ with $\gamma_m = 0.05$, for the RAS and the SCH functions corresponds to that for the OneMax problem, as shown in Fig. 4.20(b).

The above results show that our theoretical arguments about MPGAs in the OneMax problem domain still hold for the four test functions. Thus it verifies the applicability of our theoretical arguments to other problem domains. Furthermore, according to the analytical results in the OneMax problem, prolonging the running time (> 10^6 generations) is expected to reinforce the superiority of MPGAs using more than two parents in OB-Scan over GAs using uniform crossover on the four test functions.



(a) RAS, $t = 10^4$

(b) RAS, $t = 10^6$

(c) SCH, $t = 10^4$

(d) SCH, $t = 10^6$

Fig. 4.34. Comparison of the mean and the standard error of the fitness between $t = 10^4$ and $t = 10^6$ on the RAS and the SCH functions

4.7 Summary

This chapter analyzed the behavior of MPGAs through Markov chain theory. Specifically, we probed into MPGAs using U-Scan and OB-Scan — both are multi-parent generalizations of uniform crossover. First, we analyzed the gene frequency altered by selection, crossover, and mutation individually. The integral effect of these operators on the gene frequency is further formulated by Markov chain theory. Using this Markov model, we examined two aspects of MPGAs: the *genetic drift* and the *convergence.*

In terms of genetic drift, the mean convergence time of MPGAs *without* selection pressure and mutation is concerned. The proposed Markov model turns out to be an exact model under the assumptions of zero selection pressure. The analytical results from this model demonstrate that the number of parents in U-Scan exercises no influence on genetic drift. In addition, the theoretical analysis shows that the number of parents in OB-Scan plays an important role in the rate of genetic drift: Raising parents in OB-Scan shortens the mean convergence time; that is to say, it accelerates genetic drift. This outcome reconfirms Schippers' claim about the tendency of genetic drift for scanning crossover; furthermore, our analysis gives the expected time of convergence. Apart from the rate of genetic drift, the probability of convergence to allele 1 shows that raising parents in OB-Scan will intensify the preference of initialization for allele 1 or 0. These theoretical results are validated by a series of experiments.

As for the convergence of MPGAs *with* selection pressure and mutation in the OneMax problem, the theoretical analysis shows several interesting points:

1. The selection intensity can significantly improve the mean convergence fitness for $n = 2$ but has little impact on the mean convergence fitness for $n > 2$. Additionally, increasing selection intensity accelerates the progress of fitness and then gains a higher mean fitness for a short running time.
2. The number of parents in U-Scan exerts no influence on the convergence of MPGAs in terms of both fitness and convergence time.
3. The number of parents in OB-Scan has a great influence on the performance of MPGAs: First, there exists the *pairwise equivalence* in OB-Scan. That is, for an even number n, the performance of n-parent OB-Scan is equivalent to that of $(n-1)$-parent OB-Scan in terms of mean fitness as well as convergence time. Second, increasing the number of parents in OB-Scan *with* mutation gains better mean convergence fitness, but at the cost of convergence time.
4. Mutation is crucial for the use of more than two parents in MPGAs using OB-Scan. The theoretical results show that MPGAs using more than two parents in OB-Scan together with mutation, on the one hand, are capable of higher mean convergence fitness than GAs using 2-parent OB-Scan, viz uniform crossover. On the other hand, using more parents in OB-Scan needs more generations to achieve convergence, especially for small mutation rate. However, the defects do not outweigh the merits. Even in a short running time, MPGAs using OB-Scan with fewer than 10 parents can achieve higher fitness than those with 2 parents. In addition, the analysis reveals

that the mutation rate should be duly increased with the number of parents in OB-Scan.

5. In the way of random survivor, the generation gap exercises little impact on the performance.
6. The population size has inconsiderable influence over mean convergence fitness but drastically increases the mean convergence time. However, a large population size holds the advantage of higher mean fitness for a short running time.

These theoretical results are verified by a series of experiments in the OneMax problem. It shows that theoretical and experimental results fit very well. This consistency validates the capability of our proposed model. Moreover, the superiority of n-parent ($n > 2$) OB-Scan over 2-parent OB-Scan in the OneMax problem is verified theoretically as well as empirically.

We further extended the analysis to the Generalized OneMax problem. Considering the independent character of the gene frequency in this class of problems, we extended the Markov model for single locus to a model for multiple loci. The convergence of MPGAs in two small instances of the Proportionate OneMax problem and the BinInt problem are investigated accordingly. Not only does the remarkable consistence between the theoretical results and the experimental results reconfirm the power of the proposed Markov model, but these results also demonstrate the advantage of MPGAs using OB-Scan over conventional GAs. The experiments on the F2, RAS, SCH, and GRI functions, furthermore, verified the applicability of our theoretical arguments to other problem domains.

In summary, the proposed model establishes a theory concerning the separate as well as the integral influences of population size, selection intensity, the number of parents, mutation rate, and generation gap on the performance of MPGAs. The theoretical results show the power and the limit of scanning crossover, and manifest the key role of mutation in the multi-parent crossover. Since scanning crossover is a generalization of uniform crossover, the theoretical results and analytical methods proposed in this work are also applicable to conventional 2-parent GAs with uniform crossover.

5

Conclusions

5.1 Summary

Multi-parent genetic algorithms (MPGAs) are genetic algorithms (GAs) using two or more parents. Traditionally, GAs adopt two parents in crossover to reproduce offspring. Multi-parent genetic algorithms break through this limitation by allowing more than two parents participating in crossover. However, issues arise from the increase of parents about the operation of crossover with more parents, about the number of parents, about the mating of them, about the situation to use this number, and about the whys and the wherefores of using it. Even though a considerable number of studies have shown the effectiveness and advantages of MPGAs, they concentrate on design of multi-parent crossovers in general. Various issues about MPGAs are still left open: What is the suitable number of parents for a given problem? Who should be mated? When will MPGAs outperform GAs? Why do MPGAs perform better?

The objective of this thesis is to resolve these issues by design and analysis of MPGAs. In Chapter 2, we designed a novel mating strategy to deal with the mating issue and the number of parents in MPGAs. The proposed approach, called the *tabu multi-parent genetic algorithm* (TMPGA), integrates the tactics of tabu search (TS) into the mating of MPGAs. TMPGA sifts parents according to the tabu restriction and the aspiration criterion of TS; consequently, this method balances the population diversity and selection pressure of MPGAs. The resultant validity of mating is further used to adjust the number of parents adaptively. Experimental results show that TMPGA outperforms GA and MPGA in terms of convergence speed and solution quality. This superiority of TMPGA validates the effectiveness of the proposed mating strategy. Moreover, we examined the correlation between the number of parents and the numbers of tabu events and aspiration events in the course of evolution. The experimental results demonstrate that the validity of mating can afford to reflect the level of population diversity and then to be a satisfactory reference for adjusting the number of parents.

In addition to design, we conducted theoretical analyses to address the issues of the situations and the reasons that MPGAs can outperform GAs. In Chapter 3, we analyzed the performance of *uniform scanning crossover* (U-Scan) and *occurrence based scanning crossover* (OB-Scan) — both are multi-parent generalizations of *uniform crossover.*

In this chapter a specific population model, called *uniform population*, was proposed as a systematic population environment for analysis. A k-order uniform population is defined as a population in which the chromosomes all have the same distance, k genes, from the unique optimal solution. A criterion based on uniform population is further presented to evaluate the performance of crossover. This analysis proved that the number of parents exercises no influence upon the performance of U-Scan; in other words, an n-parent U-Scan with $n > 2$ will perform identically with its 2-parent version, i.e. uniform crossover. As for OB-Scan, the analysis revealed the pairwise equivalence phenomenon: OB-Scan with an even number n of parents performs equivalently to OB-Scan with $n-1$ (odd number) parents. Depending on the analytical results about the probability for OB-Scan to reproduce better, equal, or worse offspring, we made a conjecture that raising parents in OB-Scan will intensify the exploitation of the evolutionary search. The analytical results from this simplified population model shows a high level of consistency with the experimental results on four thorny test functions, which validates the capability of this analysis.

Beyond the assumption of uniform population, in Chapter 4 we used Markov chain theory to model the behavior of MPGAs and to analyze the number of parents, the parameter setting, and the whys and the wherefores of the superiority of MPGAs over GAs. First, we looked into the variation of gene frequency affected by selection, multi-parent crossover, and mutation individually. Next, we formulated by Markov chain theory the integral influence of the MPGA operators, including survivor strategy, on the variation of gene frequency over generations. Accordingly, the proposed Markov model concerns the separate as well as the integral effects of the *population size*, the *selection intensity* in selection, the *number of parents* in crossover, the *mutation rate* in mutation, and the *generation gap* in survivor over generations.

Two aspects of MPGAs were investigated in Chapter 4: the *genetic drift* and the *convergence*. In terms of genetic drift, we examined the principal measure of the rate of genetic drift — the mean convergence time of MPGAs *without* selection and mutation. In this case, the proposed Markov model turns out to be an exact model for the rate of gene drift in MPGAs. The analytical results indicated the number of parents in U-Scan has no influence while that number in OB-Scan plays an important role in the genetic drift of MPGAs: Raising parents in OB-Scan accelerates the genetic drift of MPGAs. Furthermore, we analyzed the probabilities for MPGAs using OB-Scan to converge to allele 0 and allele 1. This theoretical analysis showed that raising parents in OB-Scan will intensify the preference of initialization for allele 0 or 1. This outcome verified the conjecture in Chapter 3 about the intensification of exploitation in MPGAs.

In terms of the convergence of MPGAs, we investigated the complete MPGAs (i.e. *with* selection and mutation) in the OneMax problem and further the Generalized OneMax problem. Several interesting points emerge from this theoretical analysis. First, we reconfirmed the arguments made by the analysis based on uniform population: The number of parents indeed exerts no influence on MPGAs using U-Scan, with regard to mean fitness and mean convergence time. In addition, the *pairwise equivalence* phenomenon in OB-Scan was proved. Second, the proposed Markov model indicated the correlation of the number of parents with the mean fitness and the mean convergence time of MPGAs using OB-Scan. Third, the analytical results manifested the critical role of mutation in the performance of MPGAs using OB-Scan. In the Generalized OneMax problems, including the OneMax problem and the BinInt problem, the theoretical results showed that MPGAs using more parents in OB-Scan with mutation, on the one hand, are capable of higher mean convergence fitness than MPGAs using 2-parent OB-Scan, viz GAs using uniform crossover. On the other hand, they need more generations to achieve convergence. However, the defects do not outweigh the merits. For a reasonably short running time, MPGAs using n-parent OB-Scan outperform GAs concerning the solution quality in the Generalized OneMax problems. More importantly, all these theoretical results were verified by a series of experiments. The good fit between the theoretical and the experimental results not only validates the soundness of the theoretical analysis but also demonstrates the effectiveness of the proposed Markov model.

5.2 Contributions

In regard to the question "Can GAs benefit from the increase of parents?", the present thesis stated that:

Yes, GAs can benefit from the increase of parents *conditionally.*

Centering around the above statement, we proposed designs to improve the performance of MPGAs and carried out analyses to characterize, describe, and predict the behavior of MPGAs in order to discover and prove the conditions that benefit MPGAs. More specifically, the contributions of this thesis can be categorized into theoretical aspect and practical aspect as follows.

1. Contributions in theoretical aspect:
   - **Theoretical Foundations:** This thesis proposed the uniform population model for theoretical analysis of crossover, investigated the variation of gene frequency caused by each MPGA operator, and developed a Markov model for MPGAs. First, the proposed uniform population affords a simple yet effectual model for theoretical analysis of crossover. The analysis based on uniform population gives the probability for a crossover to reproduce better, equal, or worse offspring; therefore, it shows the effectiveness/ineffectiveness of a given crossover. The theoretical results from this analysis are greatly consistent with the experimental results and with the theoretical results from Markov chain theory. Second, we built a theoretical foundation of the analysis on the variation of gene frequency caused by selection, multi-parent crossover, and mutation. It suffices to specify the respective effects of these operators on gene frequency. Third, a Markov model was constructed to formulate the integral influence of MPGA operators on the gene frequency in the course of evolution. This model is capable of describing and predicting the behavior of MPGAs. These theoretical analyses are

all examined by a series of experiments. The good fit between the theoretical results and the experimental results validates the correctness and the effectiveness of these theoretical analyses.

- **Characterization of Raising Parents in MPGAs:** Two characters of MPGAs were explored: the *exploitation* and the *genetic drift*. The analysis based on uniform population clued the impact of multi-parent crossover on exploitation of the evolutionary search. Moreover, this thesis gave a theoretical analysis on the genetic drift of MPGAs. This analysis explicitly indicates the expected time (generations) for the population diversity to drain away and the probability for MPGAs to drift into allele 1 or allele 0. The theoretical results showed that the increase of parents in OB-Scan on the one hand will intensify the exploitation and, on the other hand, will accelerate the genetic drift of MPGAs. For MPGAs using U-Scan, this thesis proved that the adopted number of parents has no effect on both exploitation and genetic drift.
- **Predictive Model for MPGAs:** We built a Markov model for MPGAs, which concerns the integral influence of the *population size*, the *selection intensity* in selection, the *number of parents* in crossover, the *mutation rate* in mutation, and the *generation gap* in survivor over generations. This model was verified in an empirical manner and successfully showed its power in predicting the mean fitness over time and the mean convergence time of MPGAs. Through this model, we are able to discover the optimal setting concerning population size, selection intensity, the number of parents, mutation rate, and generation gap in MPGAs.

2. Contributions in practical aspect:
   - **Improvement of Performance:** A mating strategy was proposed for MPGAs to sift parents in consideration of the balance between population diversity and selection pressure. This mating strategy further adjusts the number of parents in response to the diversity in the population. The MPGA using this mating strategy was shown empirically to outperform GA as well as the MPGA without this mating strategy. Moreover, the results from our theoretical analyses suggested several guidelines to improve the solution quality and convergence speed of MPGAs.
   - **Guidelines for Determining the Number of Parents:** We achieved this goal by means of design and analysis of MPGAs. First, a rule was proposed to adaptively adjust the number of parents according to the validity of mating. Second, the theoretical analyses gave several useful hints on determining the number of parents: For MPGAs using U-Scan, we suggested adopting 2-parent U-Scan, namely uniform crossover, in that both the analysis based on uniform population and the analysis based on Markov chain theory pointed to the same conclusion — the number of parents exerts no influence on the performance of MPGAs using U-Scan. In that case, raising parents in U-Scan receives no gains but the cost of selecting extra parents. Thus, two parents are suggested in

U-Scan. For MPGAs using OB-Scan, this thesis suggested using only odd numbers of parents (except $n = 2$) owing to the pairwise equivalence phenomenon indicated in our theoretical analyses. In addition, in terms of solution quality and convergence speed, the conjecture "the more parents, the better" conditionally holds in OB-Scan — it strongly depends on the parameter setting of other operators.

- **Guidelines for Determining the Parameters of Other Operators:** The proposed Markov model is able to predict the behavior and to find the optimal setting of MPGAs in the Generalized OneMax problem. For MPGAs using OB-Scan, this thesis revealed that the utilization of mutation has a significant influence on the performance of MPGAs. In addition, we argued that the mutation rate should be duly increased with the number of parents in OB-Scan to enhance solution quality and to prevent premature convergence, because of the intensified exploitation and the accelerated genetic drift by raising the number of parents in OB-Scan. For MPGAs using U-Scan, we proved that n-parent U-Scan with $n \in \mathbb{N}_{>1}$ corresponds to 2-parent OB-Scan. Hence, for the parameter setting of MPGAs using n-parent U-Scan one can refer to that of MPGAs using 2-parent OB-Scan.

5.3 Future Work

In this thesis, we designed a mating strategy to deal with the mating issue and the number of parents in MPGAs, and have received preferable results in solution quality and convergence speed. According to the tabu restriction, this mating strategy filters parents in an incest-prevention manner. That is to say, this strategy encourages the mating of distant parents, which can be viewed as a memory-based implementation of *heterosis* in MPGAs as well as GAs [107]. Even though heterosis has its advantage in nature [42, 53] and the mating strategies based on it received many success stories in GAs [37, 107], it is still lacking in theoretical demonstration of the benefits to use these heterosis-like approaches.

The uniform population model is capable of a simple yet effectual model for analysis of crossover. This thesis used it to derive the probability for a crossover to reproduce better or worse offspring. Several extensions to broaden the usage of this model still remain for future study: In terms of the performance criterion associated with uniform population, we merely aimed for the case of performance equivalence. The probe into other cases concerning the combinations of k-positive, k-ineffective, and k-negative probabilities is expected to clue about an operator's strength in exploitation and exploration. Furthermore, in the analysis we focused on the progress of uniform population towards its subsequent generation. The *evolution* was not taken into account; nevertheless, it is of interest to look into the subsequence of uniform population. The results from this extension can further afford to show the performance of crossover in the long term.

The present thesis analyzed in theory the variation of gene frequency caused by respective MPGA operators, and further formulated the integral influence of these variations over generations by Markov chain theory. We conducted an analysis on two scanning crossovers, U-Scan and OB-Scan. These two multi-parent crossovers generalize the classic uniform crossover, while uniform crossover generalizes other common 2-parent crossovers: one-point crossover, two-point crossover, and multi-point crossover. However, we cannot directly apply the theoretical results of scanning crossover to diagonal crossover or multi-cut crossover, which are multi-parent generalizations of one-point crossover and two-point crossover, respectively. The reason is that the operation of diagonal crossover or multi-cut crossover induces *linkage* of genes in a parent, which is fundamentally different from the *bitwise* operation of scanning crossover. An investigation into this linkage factor in the multi-parent crossovers would be challenging yet fruitful for the advanced knowledge about the role of parents in MPGAs.

As for the problem domains of analysis, this thesis focused on the OneMax problem and the Generalized OneMax problem. Even though these problems are relatively simple, the analysis for them is applicable to other problem domains. The analysis presented in this thesis has successfully identified the influence of raising parents on MPGAs using U-Scan and OB-Scan. On the other hand, since the mechanism of OB-Scan is based on majority voting, it seems vulnerable to the deceptive or trapping problems. The monotonic fitness of the Generalized OneMax problem, therefore, cannot afford to investigate this vulnerability in theory. An advanced analysis on other problem domains, especially the deceptive problems, will be of great help for discovering the behavior of MPGAs using, but not restricted to, OB-Scan.

The last but not the least direction to extend this thesis is practical application of MPGAs. Genetic algorithms are devised to deal with problems which have no satisfactory solution yet. Commonly GAs are used as an optimization algorithm, for example, in machine learning [48, 69] and combinatorial problems [51, 82]. Recently, in the EC community it is popular to apply evolutionary algorithms to handle problems in bioinformatics [41, 84], computer security and cryptology [14, 67], and art [81, 84]. The present thesis showed empirically and theoretically the superiority of MPGAs over GAs, and provided the guidelines to choose parameters and to improve MPGAs. Accordingly, MPGAs have a great potential to outperform GAs — It is more than promising and interesting to apply MPGAs to solve practical problems in place of GAs.

A

Test Functions

A.1 De Jong's Function 2 (F2)

De Jong's F2 function (also known as the Generalized Rosenbrock's function) [20] can be generalized by

$$f_{\mathrm{F2e}}(\mathbf{x}) = \sum_{i=1}^{N-1} \left[\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2 \right],$$

where $-2.048 \leq x_i \leq 2.048$ and $N \in \mathbb{N}_{>1}$. This function is a continuous, unimodal, non-convex, and N-dimensional quadratic function with a minimum of 0 at $\mathbf{x} = (1, \ldots, 1)$. The deep parabolic valley of F2, as shown in Fig. A.1, causes a severe difficulty in finding the global minimum.

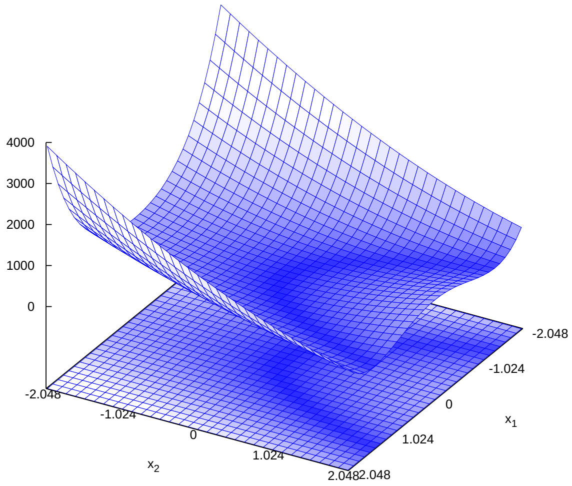


Fig. A.1. Fitness landscape of F2

A.2 The Rastrigin Function (RAS)

The Rastrigin function is given by

$$f_{\mathrm{RAS}}(\mathbf{x}) = 10N + \sum_{i=1}^{N} \left[x_i^2 - 10\cos(2\pi x_i) \right] ,$$

where $-5.12 \leq x_i \leq 5.12$ and $N \in \mathbb{N}$. This function is a continuous, multimodal, nonconvex, and N-dimensional function with a minimum of 0 at $\mathbf{x} = (0, \ldots, 0)$. As Fig. A.2 shows, the RAS function has many and widespread local minima, which hinder search algorithms from finding the global minimum.

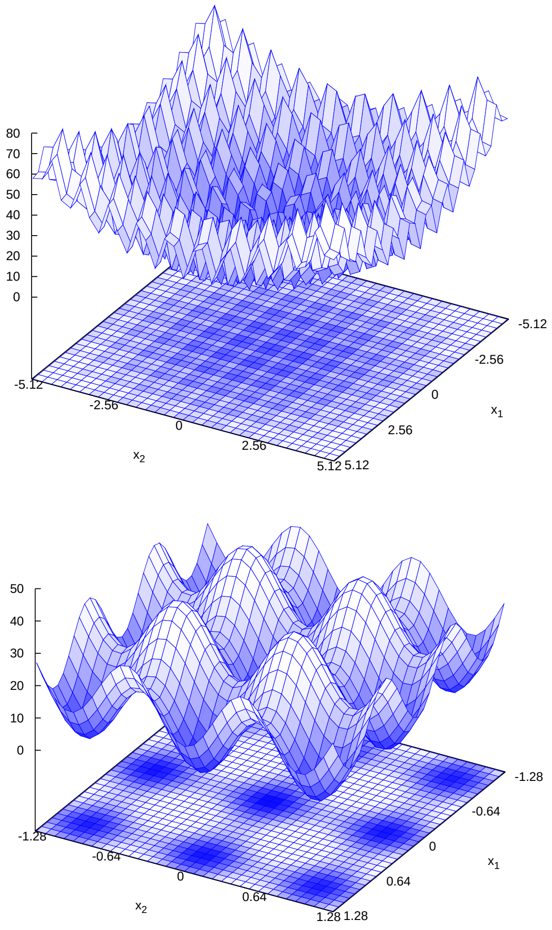


Fig. A.2. Fitness landscape of RAS

A.3 The Schwefel Function (SCH)

The Schwefel function is given by

$$f_{\mathrm{SCH}}(\mathbf{x}) = 418.98291N - \sum_{i=1}^{N} x \sin\left(\sqrt{|x_i|}\right),$$

where $-512 \le x_i \le 512$ and $N \in \mathbb{N}$. This function is a continuous, multimodal, non-convex, and N-dimensional function with a minimum of 0 at $\mathbf{x} = (420.9687, \ldots, 420.9687)$. In addition, the SCH function is deceptive since its second-best minimum is distant from the global minimum; thus the search algorithms are potentially directed in the wrong way. The landscape of a 2-dimensional SCH function is plotted in Fig. A.3.

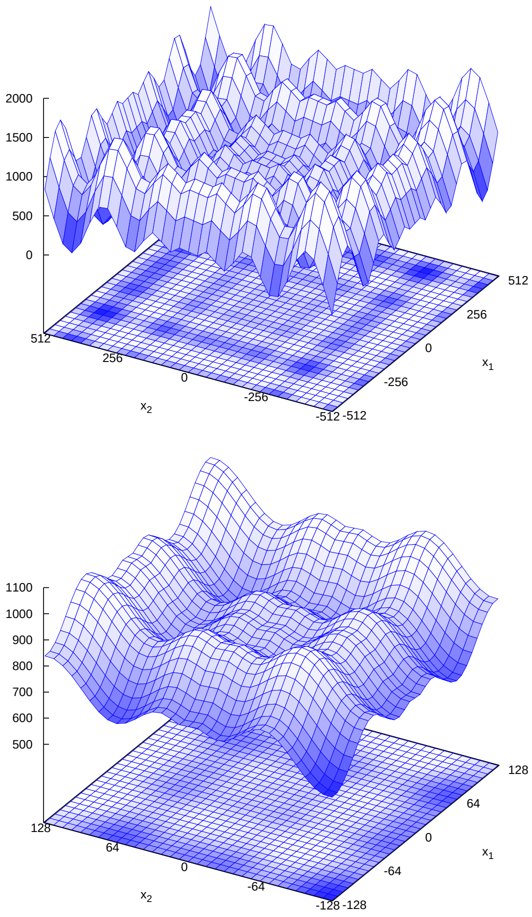


Fig. A.3. Fitness landscape of SCH

A.4 The Griewangk Function (GRI)

The Griewangk function is given by

$$f_{\mathrm{GRI}}(\mathbf{x}) = 1 + \sum_{i=1}^{N} \frac{x_i^2}{4000} - \prod_{i=1}^{N} \cos\left(\frac{x_i}{\sqrt{i}}\right),$$

where $-512 \leq x_i \leq 512$ and $N \in \mathbb{N}$. This function is a continuous, multimodal, non-convex, and N-dimensional function with a minimum of 0 at $\mathbf{x} = (0, \ldots, 0)$. The GRI function is a challenging problem because of its exponentially increasing number of local minima with the dimension N. The enlarged fitness landscape in Fig. A.2 demonstrates the vast number of local minima in a 2-dimensional GRI function.

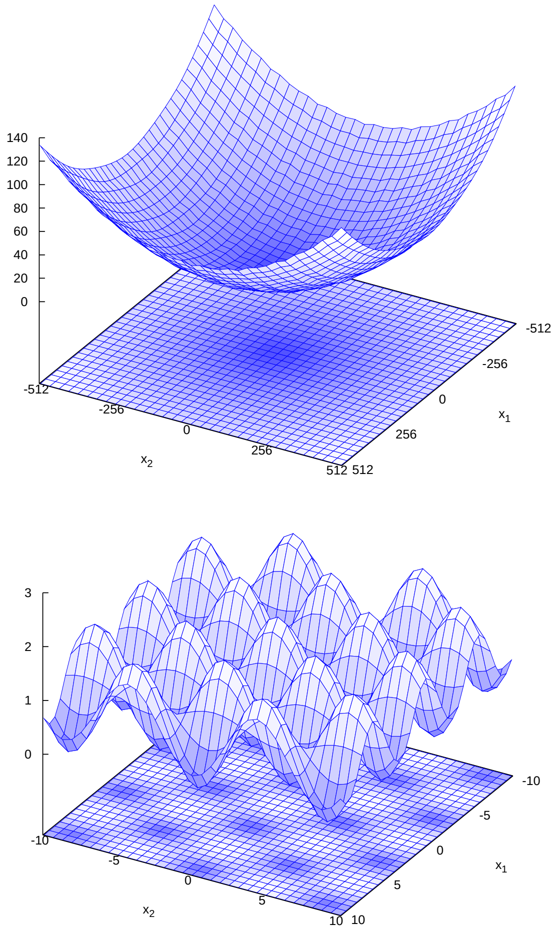
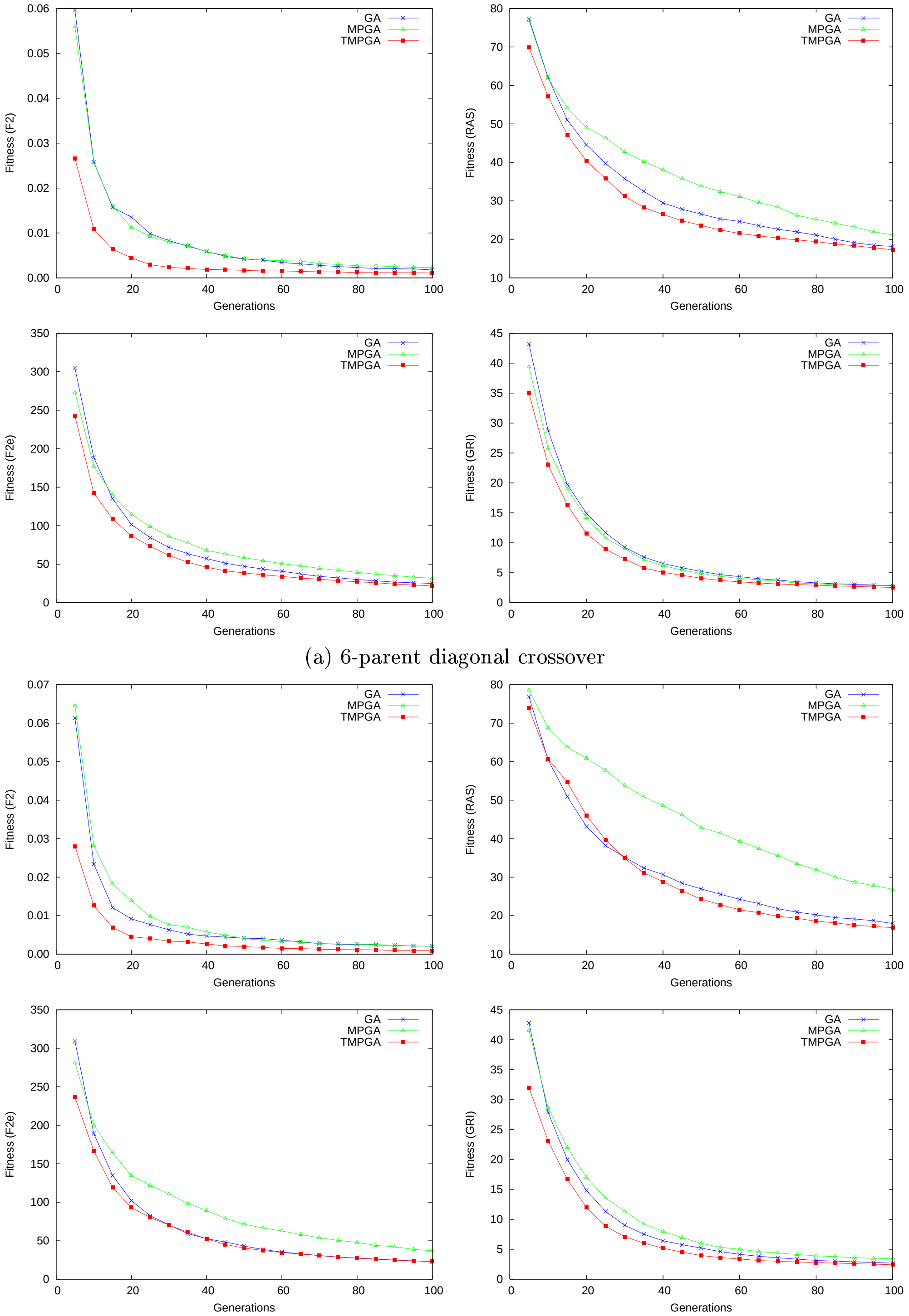


Fig. A.4. Fitness landscape of GRI

B

Experimental Results: TMPGA

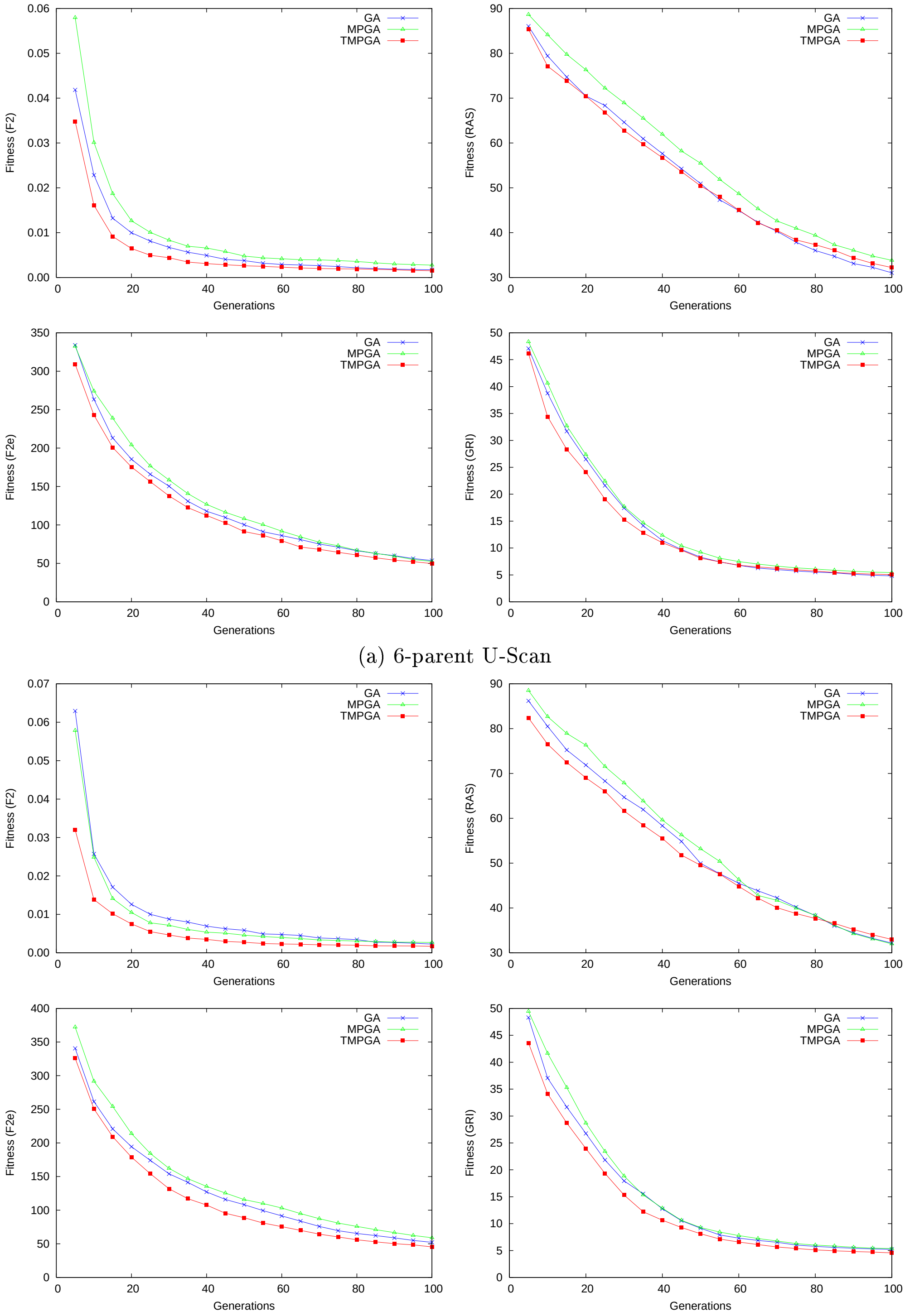
This section presents the experimental results of GA, MPGA, and TMPGA in four test functions: De Jong's second test function (F2), an extended F2 function (F2e), the Rastrigin function (RAS), and the Griewangk function (GRI). Two numbers (6 and 15) of parents for MPGA and TMPGA and two mutation rates ($\frac{1}{l}$ and 0.1) are considered. Through these experimental results, we compare the performance of GA, MPGA, and TMPGA with respect to three multi-parent crossovers: diagonal crossover, U-Scan, and OB-Scan in Section 2.4.2.



(a) 6-parent diagonal crossover

(b) 15-parent diagonal crossover

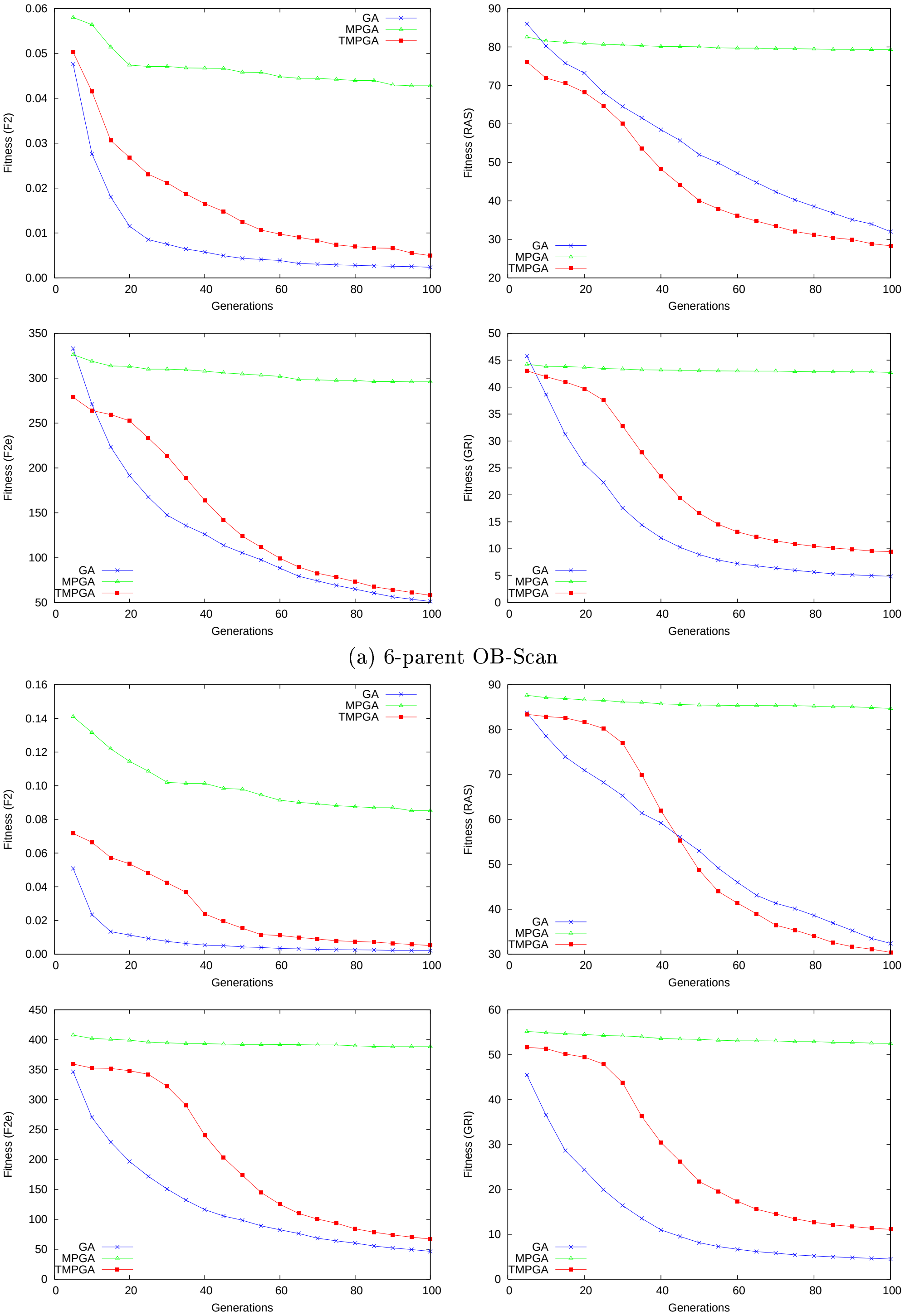
Fig. B.1. Performance comparison of GA, MPGA, and TMPGA using diagonal crossover and mutation rate $\frac{1}{l}$



(a) 6-parent U-Scan

(b) 15-parent U-Scan

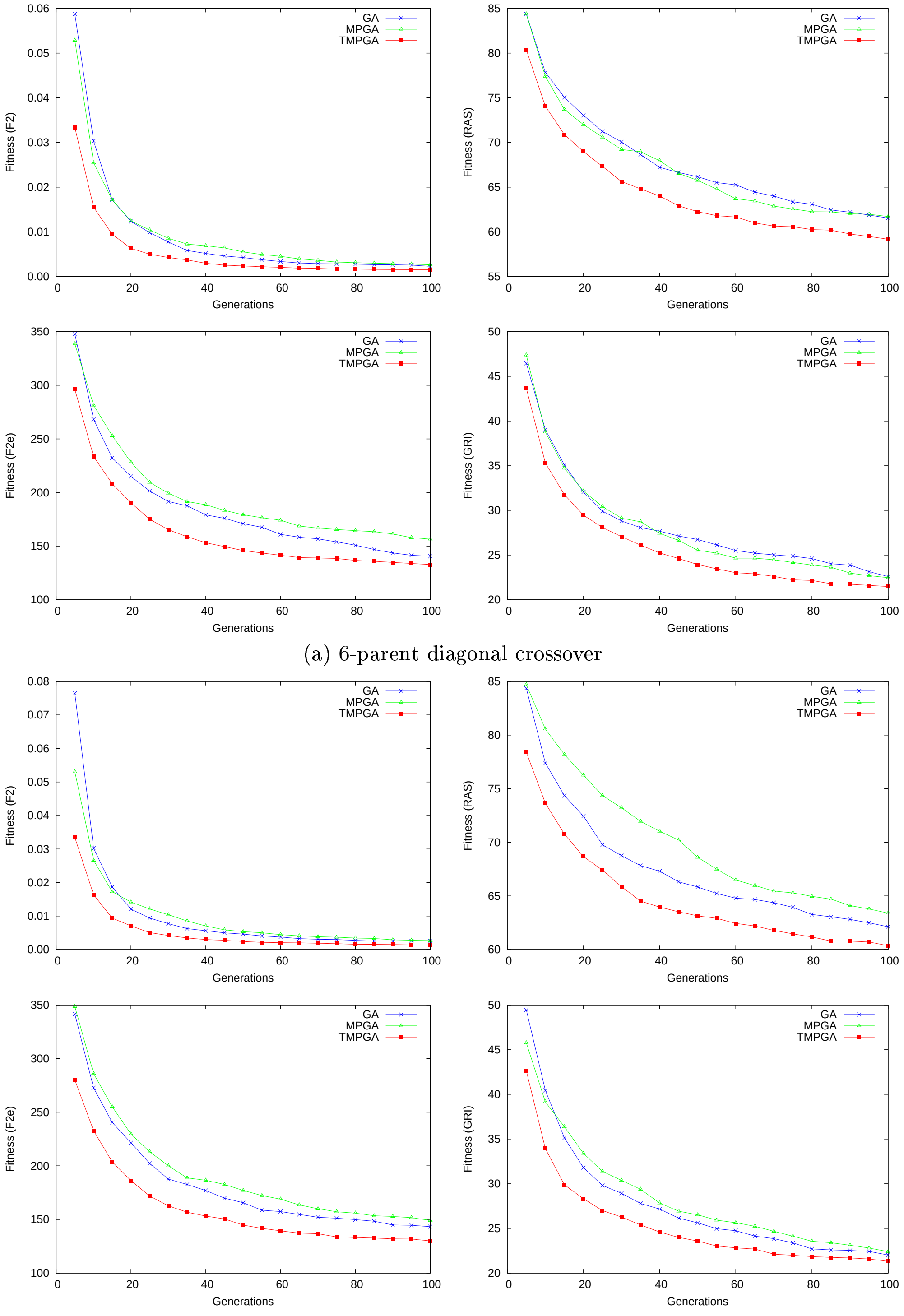
Fig. B.2. Performance comparison of GA, MPGA, and TMPGA using 15-parent U-Scan and mutation rate $\frac{1}{l}$



(a) 6-parent OB-Scan

(b) 15-parent OB-Scan

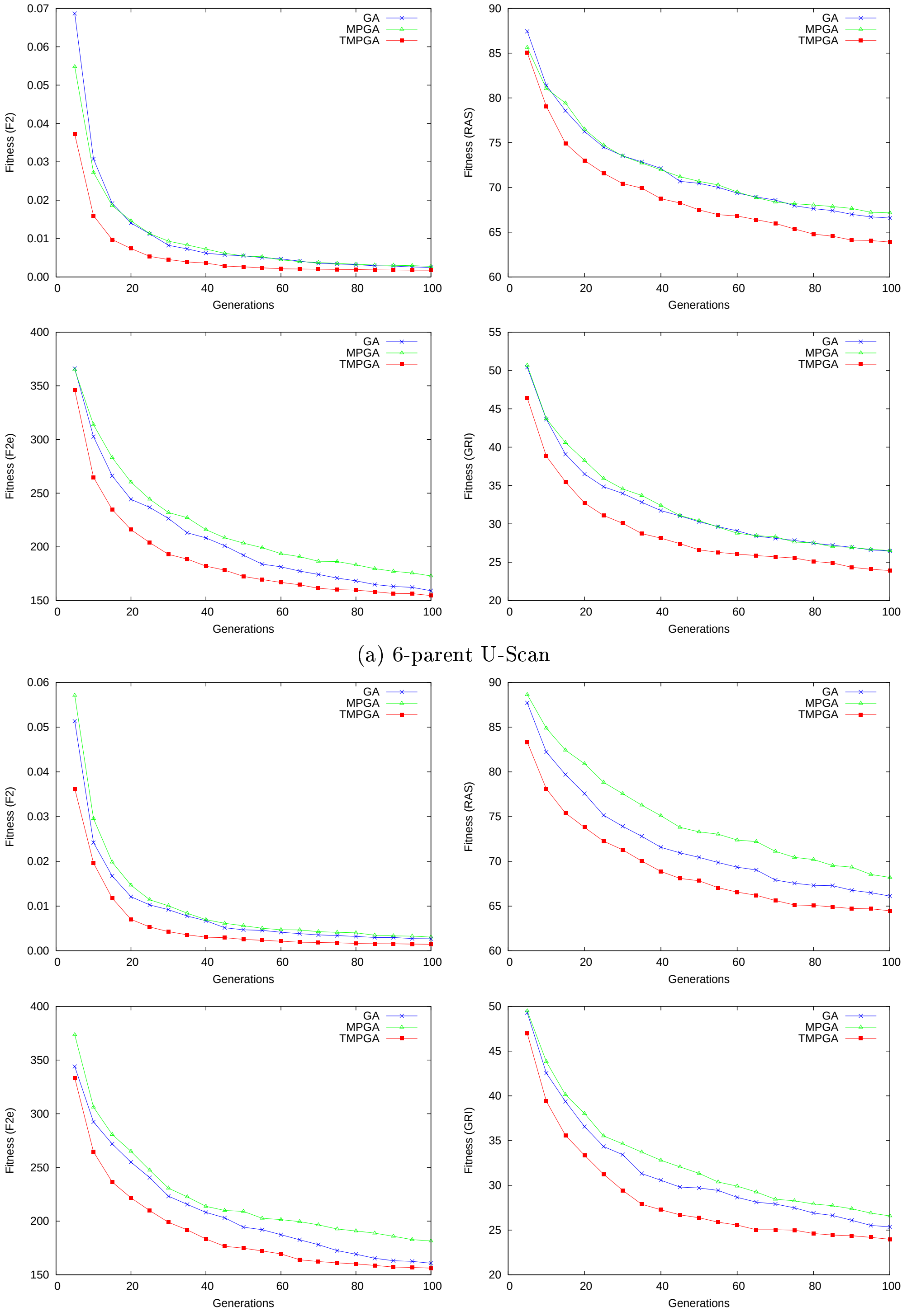
Fig. B.3. Performance comparison of GA, MPGA, and TMPGA using 15-parent OB-Scan and mutation rate $\frac{1}{l}$



(a) 6-parent diagonal crossover

(b) 15-parent diagonal crossover

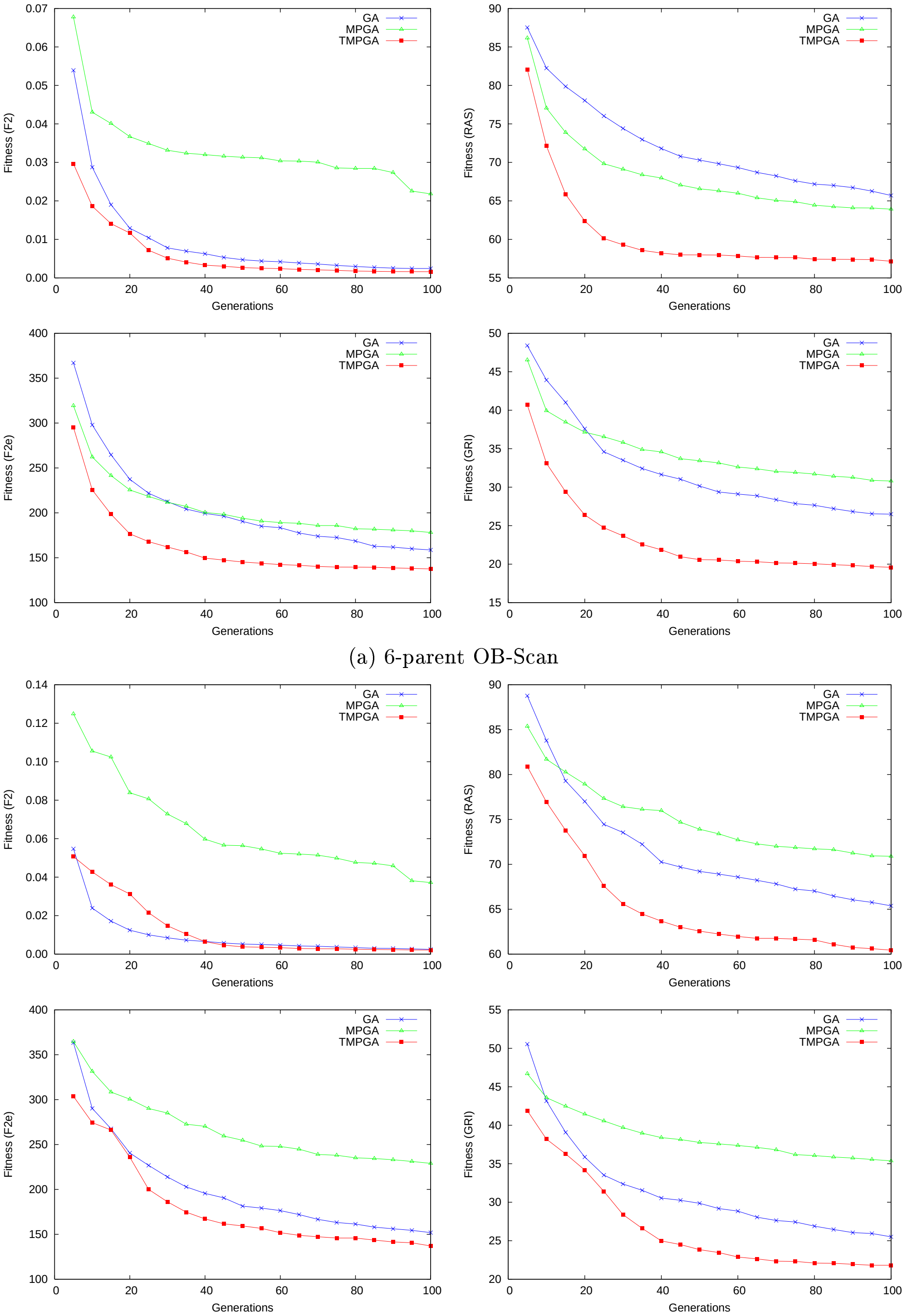
Fig. B.4. Performance comparison of GA, MPGA, and TMPGA using diagonal crossover and mutation rate 0.1



(a) 6-parent U-Scan

(b) 15-parent U-Scan

Fig. B.5. Performance comparison of GA, MPGA, and TMPGA using U-Scan and mutation rate 0.1



(a) 6-parent OB-Scan

(b) 15-parent OB-Scan

Fig. B.6. Performance comparison of GA, MPGA, and TMPGA using OB-Scan and mutation rate 0.1

C

Experimental and Theoretical Results: Convergence of MPGAs

This section presents the experimental and the theoretical results of MPGAs using OB-Scan in the Proportionate OneMax problem, the BinInt problem, the extended De Jong's second function (F2), the Rastrigin function (RAS), the Schwefel function (SCH), and the Griewangk function (GRI). The discussions of these results are presented in Sections 4.5 and 4.6.

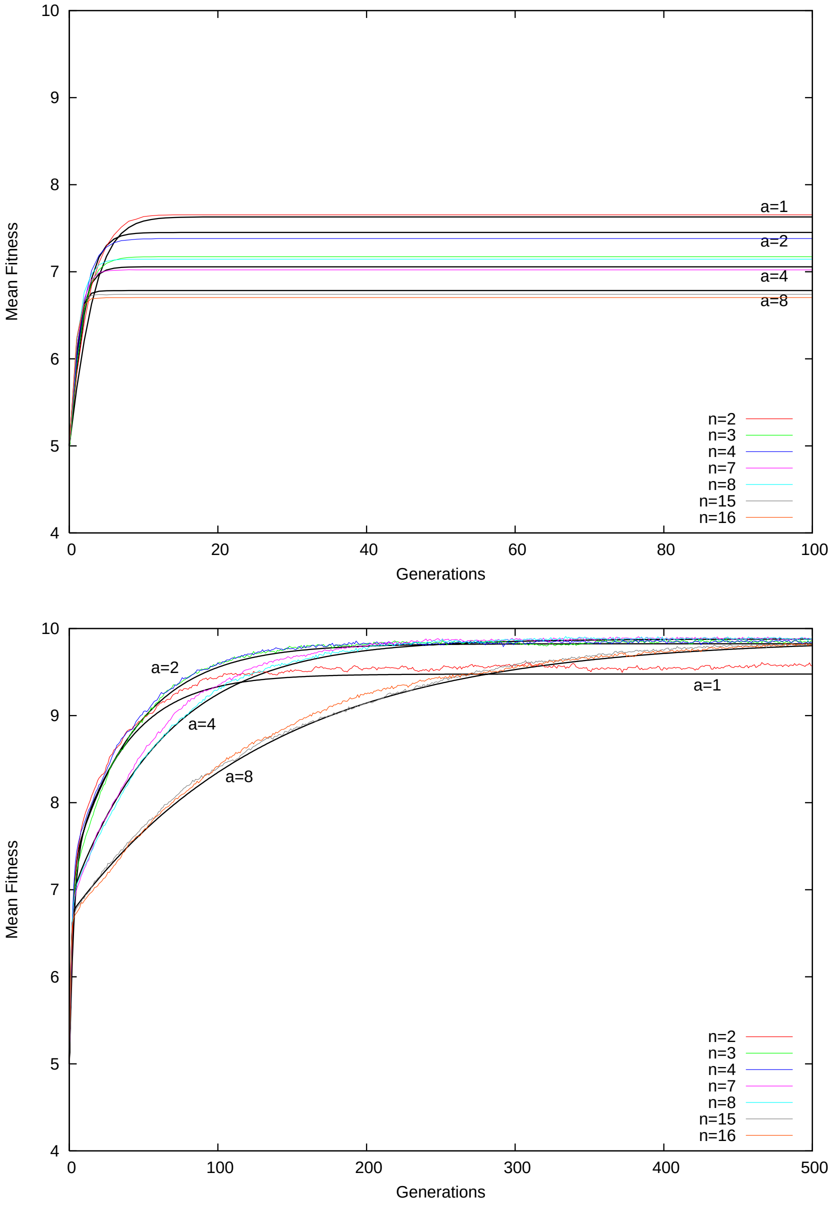
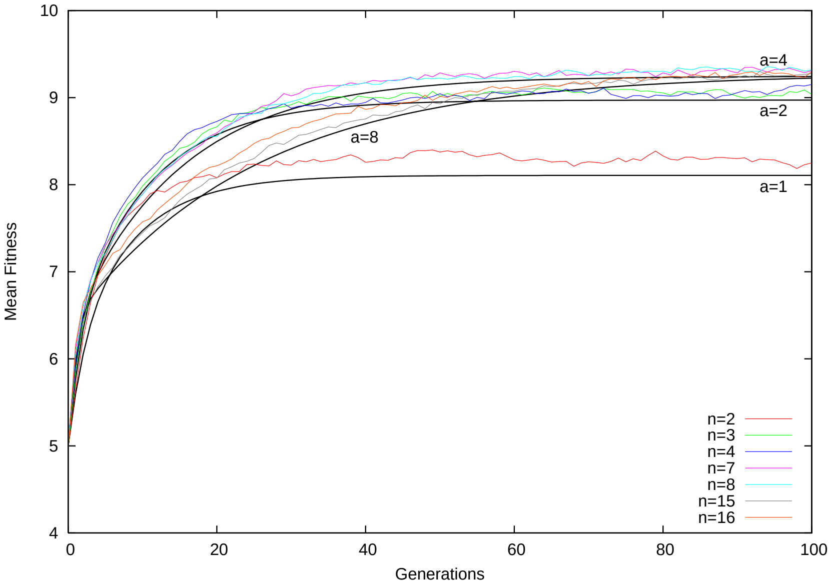


Fig. C.1. Comparison of the mean fitness obtained from theory (black bold lines) and experiment (color thin lines) in the Proportionate OneMax problem for MPGAs using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan ($a = \lceil n/2 \rceil$), and bit-flip mutation with $\gamma_m = 0$ (top) and $\gamma_m = 0.01$ (bottom)



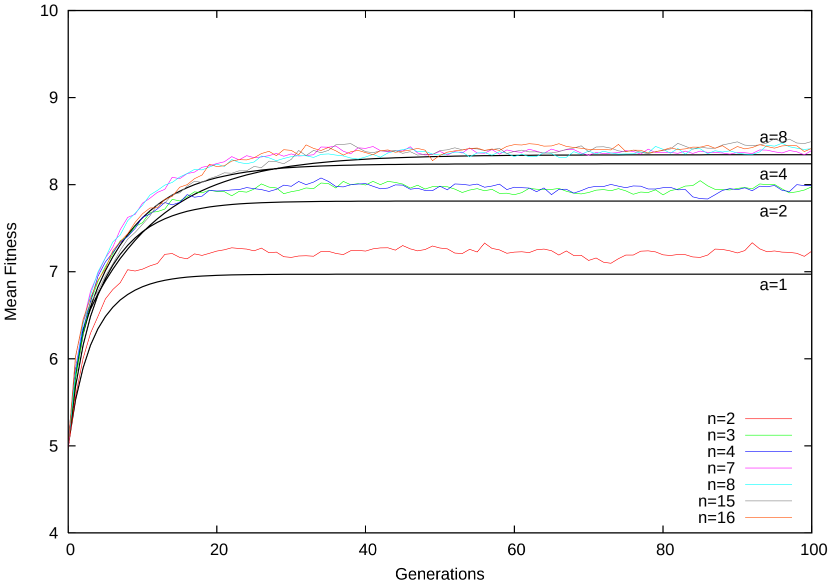


Fig. C.2. Comparison of the mean fitness obtained from theory (black bold lines) and experiment (color thin lines) in the Proportionate OneMax problem for MPGAs using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan ($a = \lceil n/2 \rceil$), and bit-flip mutation with $\gamma_m = 0.05$ (top) and $\gamma_m = 0.10$ (bottom)

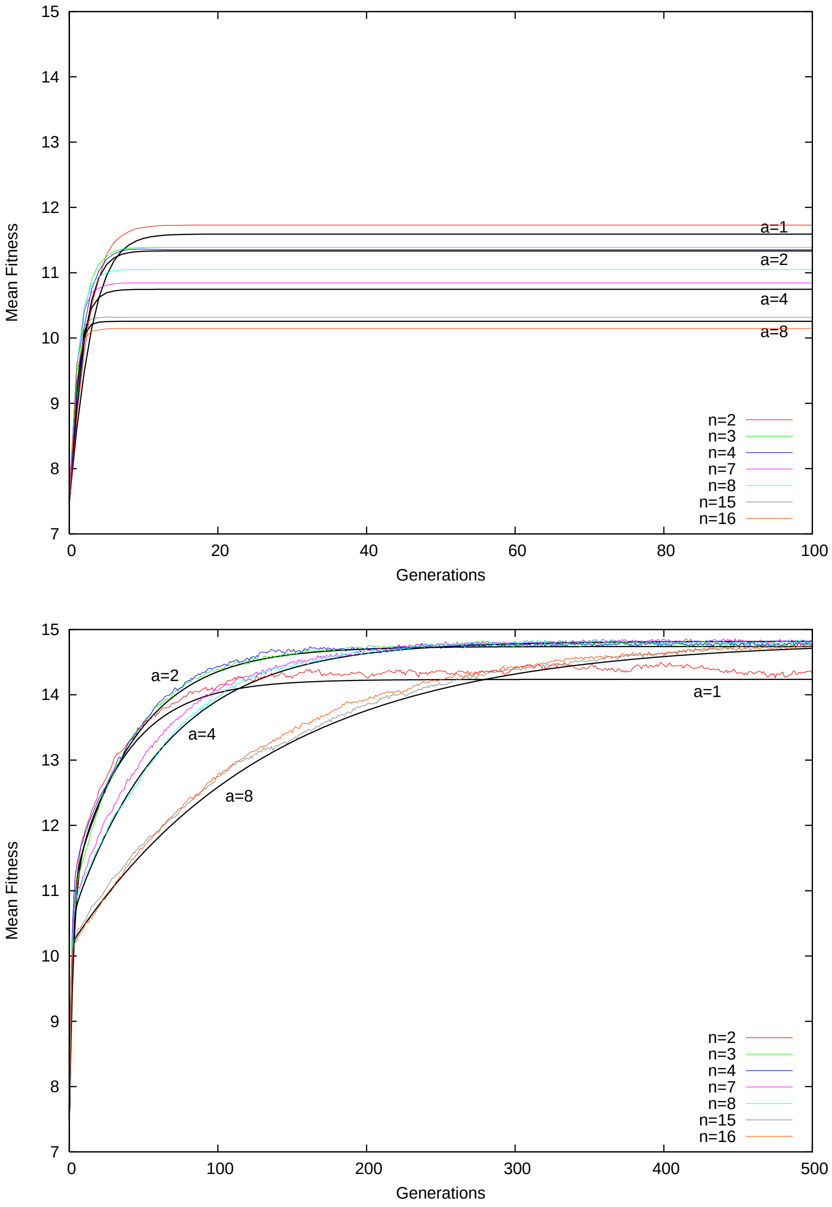
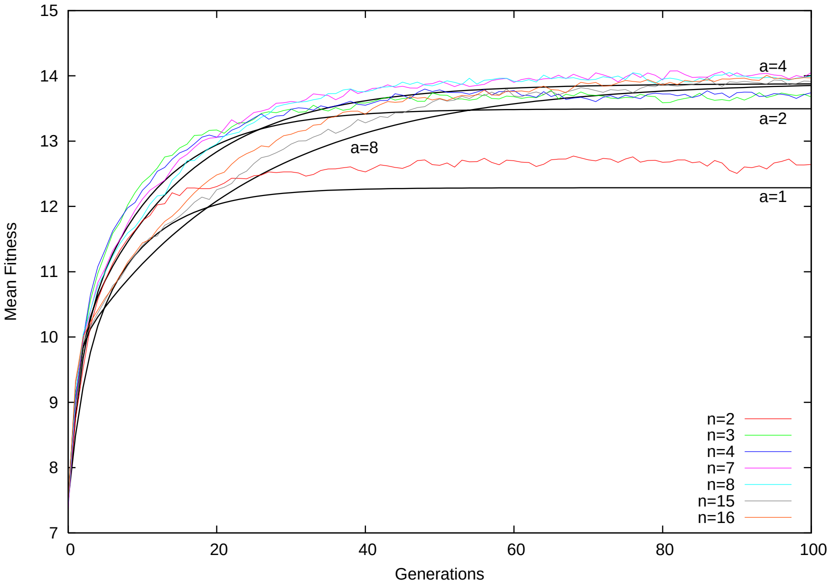


Fig. C.3. Comparison of the mean fitness obtained from theory (black bold lines) and experiment (color thin lines) in the BinInt problem for MPGAs using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan ($a = \lceil n/2 \rceil$), and bit-flip mutation with $\gamma_m = 0$ (top) and $\gamma_m = 0.01$ (bottom)



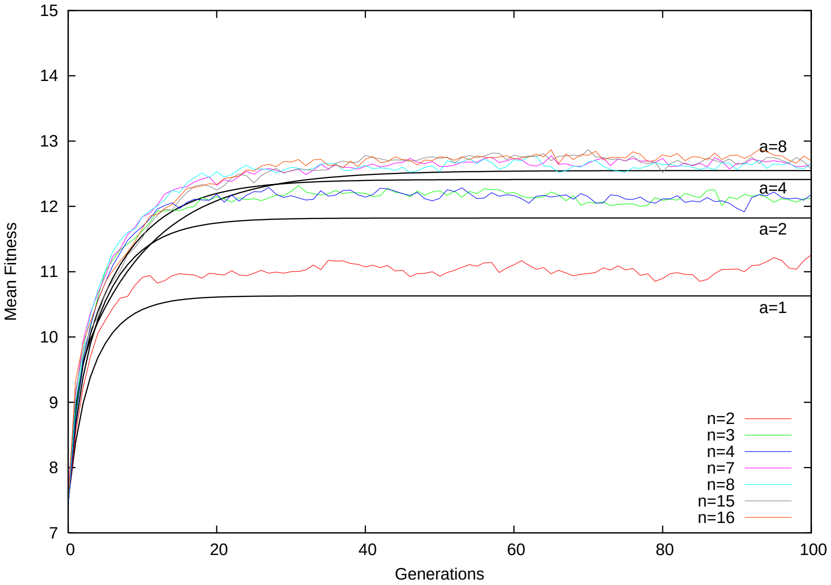


Fig. C.4. Comparison of the mean fitness obtained from theory (black bold lines) and experiment (color thin lines) in the BinInt problem for MPGAs using linear ranking selection with $\eta^+ = 1.5$, n-parent OB-Scan ($a = \lceil n/2 \rceil$), and bit-flip mutation with $\gamma_m = 0.05$ (top) and $\gamma_m = 0.10$ (bottom)

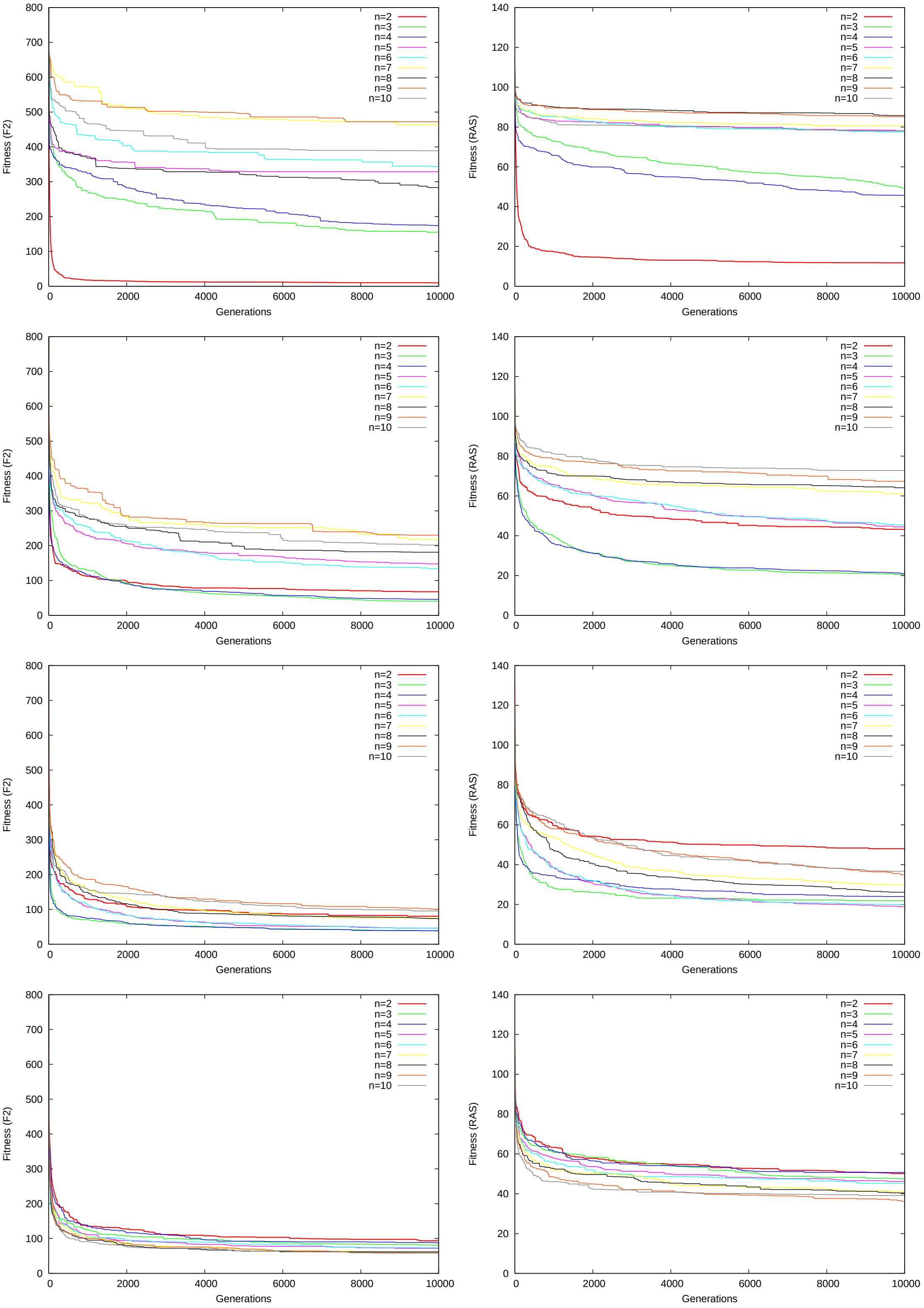


Fig. C.5. Convergence of MPGAs using n-parent OB-Scan on the F2 function (left column) and the RAS function (right column) for mutation rate $\gamma_m = 0.01$, 0.05, 0.1, and 0.2 (from top to bottom)

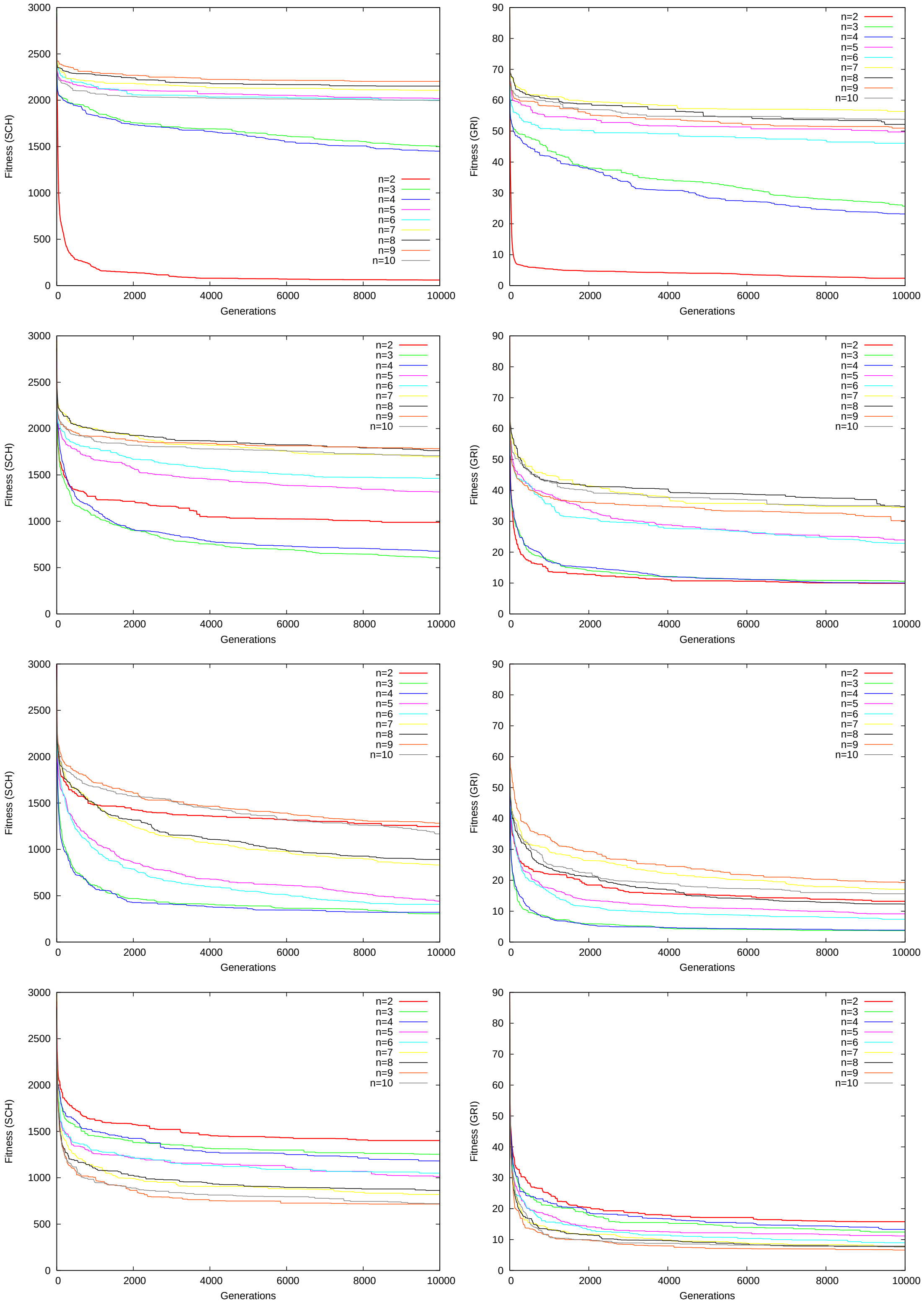


Fig. C.6. Convergence of MPGAs using n-parent OB-Scan on the SCH function (left column) and the GRI function (right column) for mutation rate $\gamma_m = 0.01$, 0.05, 0.1, and 0.2 (from top to bottom)

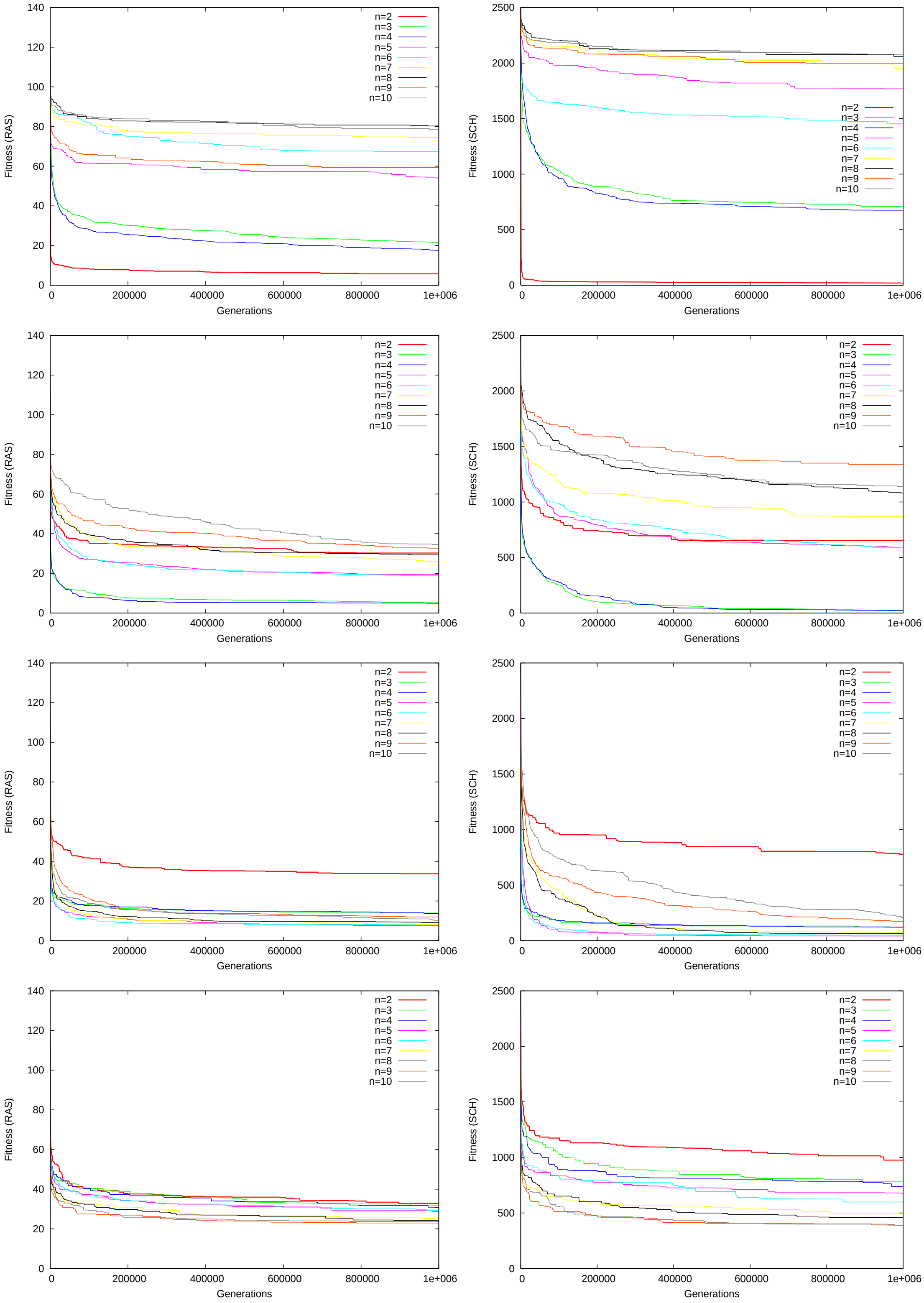


Fig. C.7. Convergence of MPGAs using n-parent OB-Scan on the RAS function (left column) and the SCH function (right column) for mutation rate $\gamma_m = 0.01$, 0.05, 0.1, and 0.2 (from top to bottom)

D

Some Proofs

D.1 Proof of Corollary 4.10

Theorem 4.43 indicated the gene frequency

$$p_k^{\mathrm{ob}(n)} = I_{p_k^s}(a,a) \quad \text{with } a = \left\lceil \frac{n}{2} \right\rceil .$$

1. $0 < p_k^s < 0.5$:
   Refer to (26.5.14) in [1]: For $p \leq 0.5$

   $$I_p(a,a) = \frac{1}{2} I_{1-p'}(a, \frac{1}{2}) \quad \text{with } p' = 4(p - \frac{1}{2})^2. \tag{D.1}$$

   The differential

   $$\begin{aligned} I_p(a+1,a+1) - I_p(a,a) &= \frac{1}{2} I_{1-p'}(a+1, \frac{1}{2}) - \frac{1}{2} I_{1-p'}(a, \frac{1}{2}) \\ &= \frac{1}{2} I_{1-p'}(a+1, \frac{1}{2}) \\ &\quad - \frac{1}{2} \left[\frac{1}{a \cdot \mathrm{Beta}(a, \frac{1}{2})} (1-p')^a (p')^{\frac{1}{2}} + I_{1-p'}(a+1, \frac{1}{2}) \right] \\ &= - \frac{1}{2a \cdot \mathrm{Beta}(a, \frac{1}{2})} (1-p')^a (p')^{\frac{1}{2}}. \end{aligned} \tag{D.2}$$

   Since for $a \in \mathbb{N}$

   $$\mathrm{Beta}\left(a, \frac{1}{2}\right) = \frac{\Gamma(a)\Gamma(\frac{1}{2})}{\Gamma(a + \frac{1}{2})} > 0$$

   and

   $$0 < p < 0.5 \Longrightarrow 0 < p' < 1,$$

   the differential for $0 < p < 0.5$

   $$I_p(a+1,a+1) - I_p(a,a) = - \frac{1}{2a \cdot \mathrm{Beta}(a, \frac{1}{2})} (1-p')^a (p')^{\frac{1}{2}} < 0,$$

which implies for $0 < p_k^s < 0.5$ and $a = \lceil \frac{n}{2} \rceil \in \mathbb{N}_{>1}$

$$I_{p_k^s}\left(\left\lceil\frac{n+2}{2}\right\rceil, \left\lceil\frac{n+2}{2}\right\rceil\right) - I_{p_k^s}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) < 0 \Longrightarrow p_k^{\mathrm{ob}(n+2)} < p_k^{\mathrm{ob}(n)}. \tag{D.3}$$

2. $0.5 < p_k^s < 1$:
   Equation (26.5.2) in [1] tells

$$I_p(a,b) = 1 - I_{1-p}(b,a). \tag{D.4}$$

   Let $q = 1 - p_k^s$. The bound $0.5 < p_k^s < 1$ implies $0 < q < 0.5$. From (D.3) and (D.4), we obtain

$$\begin{aligned}
&I_q\left(\left\lceil\frac{n+2}{2}\right\rceil, \left\lceil\frac{n+2}{2}\right\rceil\right) - I_q\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) < 0 \\
\Longrightarrow &\left[1 - I_{1-q}\left(\left\lceil\frac{n+2}{2}\right\rceil, \left\lceil\frac{n+2}{2}\right\rceil\right)\right] - \left[1 - I_{1-q}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right)\right] < 0 \\
\Longrightarrow &I_{p_k^s}\left(\left\lceil\frac{n}{2}\right\rceil, \left\lceil\frac{n}{2}\right\rceil\right) - I_{p_k^s}\left(\left\lceil\frac{n+2}{2}\right\rceil, \left\lceil\frac{n+2}{2}\right\rceil\right) < 0 \\
\Longrightarrow &p_k^{\mathrm{ob}(n)} < p_k^{\mathrm{ob}(n+2)}.
\end{aligned}$$

3. Otherwise:
   a) $p_k^s = 0.5$:
      Substitute $p = 0.5$ into (D.4) and we have for all $a \in \mathbb{N}$

$$I_{0.5}(a,a) = 1 - I_{1-0.5}(a,a) \Longleftrightarrow I_{0.5}(a,a) = 0.5$$

      Thus, as $p_k^s = 0.5$,

$$\forall n \in \mathbb{N}_{>1} : p_k^{\mathrm{ob}(n)} = I_{0.5}(a,a) = 0.5 = p_k^s.$$

   b) $p_k^s = 0$:
      As $p_k^s = 0$, for all $n \in \mathbb{N}_{>1}$ with $a = \lceil \frac{n}{2} \rceil$

$$p_k^{\mathrm{ob}(n)} = I_0(a,a) = \sum_{i=a}^{2a-1} \binom{2a-1}{i} 0^i (1-0)^{2a-1-i} = 0 = p_k^s.$$

   c) $p_k^s = 1$:
      Similarly, as $p_k^s = 1$, for all $n \in \mathbb{N}_{>1}$ with $a = \lceil \frac{n}{2} \rceil$

$$p_k^{\mathrm{ob}(n)} = I_1(a,a) = \sum_{i=a}^{2a-1} \binom{2a-1}{i} 1^i (1-1)^{2a-1-i} = 1 = p_k^s.$$

□

D.2 Proof of Corollary 4.11

1. $0.5 < p_k^s < 1$:
   We prove by Mathematical Induction (M.I.):
   $a = 2$: Since
   $$I_{p_k^s}(1,1) = \sum_{i=1}^{1} B(i; 1, p_k^s) = p_k^s,$$
   from Corollary 4.10 we have for $0.5 < p_k^s < 1$
   $$p_k^{\text{ob}} - p_k^s = I_{p_k^s}(a,a) - p_k^s = I_{p_k^s}(2,2) - I_{p_k^s}(1,1) > 0.$$
   $a = i$: Assume $I_{p_k^s}(i,i) > p_k^s$ holds.
   $a = i+1$: For $0.5 < p_k^s < 1$, from Corollary 4.10 we obtain
   $$I_{p_k^s}(i+1,i+1) > I_{p_k^s}(i,i) \Longrightarrow I_{p_k^s}(i+1,i+1) > p_k^s.$$
   According to M.I. we proved that for $0.5 < p_k^s < 1$ and $a \in \mathbb{N}_{>1}$, the inequality $I_{p_k^s}(a,a) > p_k^s$ holds, which implies for $n \in \mathbb{N}_{>2}$ the gene frequency $p_k^{\text{ob}(n)} > p_k^s$.
2. $0 < p_k^s < 0.5$:
   Let $q = 1 - p_k^s$. The bound $0 < p_k^s < 0.5$ implies $0.5 < q < 1$. From the above proof and (D.4), we have for $a \in \mathbb{N}_{>1}$
   $$\begin{aligned} I_q(a,a) > q = I_q(1,1) &\Longrightarrow 1 - I_{1-q}(a,a) > 1 - I_{1-q}(1,1) \\ &\Longrightarrow I_{p_k^s}(a,a) < I_{p_k^s}(1,1) = p_k^s. \end{aligned}$$
   Therefore, it holds for $0 < p_k^s < 0.5$ and $n \in \mathbb{N}_{>2}$ that $p_k^{\text{ob}(n)} < p_k^s$.
3. The proof of the cases $p_k^s = 0$, 0.5 and 1 refer to the third part of proof of Corollary 4.10.

□

References

1. M. Abramowitz and I.A. Stegun, editors. *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* Dover Publications, 1972. ninth Dover printing.
2. A. Agapie. Modelling genetic algorithms: From markov chains to dependence with complete connections. In *Proceedings of the 5th Parallel Problem Solving from Nature (PPSN V)*, pages 3–12. Springer-Verlag, 1998.
3. K.S. Al-Sultan and C.A. Fedjki. A tabu search-based algorithm for the fuzzy clustering problem. *Pattern Recognition*, 30(12):2023–2030, 1997.
4. L. Altenberg. The schema theorem and Price's theorem. In *Foundations of Genetic Algorithms III*, pages 23–49. Morgan Kaufmann, 1995.
5. H. Asoh and H. Mühlenbein. On the mean convergence time of evolutionary algorithms without selection and mutation. In *Parallel Problem Solving from Nature – PPSN III*, volume 866 of *LNCS*, pages 88–97, Berlin, 1994. Springer.
6. T. Bäck. Selective pressure in evolutionary algorithms: A characterization of selection mechanisms. In *Proceedings of the First IEEE Conference on Evolutionary Computation*, pages 57–62, 1994.
7. T. Bäck, D.B. Fogel, and Z. Michalewicz. *Handbook of Evolutionary Computation.* Institute of Physics Publishing and Oxford University Press, 1997.
8. T. Bäck and Z. Michalewicz. Test landscapes. In *Handbook of Evolutionary Computation*, pages B2.7:14–20. Institute of Physics Publishing and Oxford University Press, 1997.
9. T. Bäck and H.-P. Schwefel. An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, 1(1):1–23, 1993.
10. E. Behrends. *Introduction to Markov Chains.* Vieweg, 2000.
11. T. Blickle and L. Thiele. A comparison of selection schemes used in genetic algorithms. TIK-Report 11, TIK Institut fur Technische Informatik und Kommunikationsnetze, Computer Engineering and Networks Laboratory, ETH, Swiss Federal Institute of Technology, 1995.
12. P. Brémaud. *Markov Chains: Gibbs Fields, Monte Carlo Simulation, and Queues.* Springer Verlag, 1999.
13. U.K. Chakraborty, K. Deb, and M. Chakraborty. Analysis of selection algorithms: A markov chain approach. *Evolutionary Computation*, 4(2):133–167, 1997.
14. J. Clark. Nature-inspired cryptography: Past, present and future. In *Proceedings of the 2003 IEEE Congress on Evolutionary Computation*, 2003.
15. C.A. Coello Coello. A comprehensive survey of evolutionary-based multiobjective optimization techniques. *Knowledge and Information Systems*, 1(3):269–308, 1999.
16. R. Craighurst and W. Martin. Enhancing ga performance through crossover prohibitions based on ancestry. In Larry Eshelman, editor, *Proceedings of the Sixth International Conference on Genetic Algorithms*, pages 130–135, San Francisco, CA, 1995. Morgan Kaufmann.
17. C. Darwin. *The Origin of Species.* John Murray, London, 1859.

18. L. Davis. Applying adaptive algorithms to epistatic domains. In *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, pages 162–164. Morgan Kaufmann, 1985.
19. T.E. Davis and J.C. Principe. A markov chain framework for the simple genetic algorithm. *Evolutionary Computation*, 1(3):269–288, 1993.
20. K.A. De Jong. *An Analysis of the Behavior of a Class of Genetic Adaptive Systems.* PhD thesis, University of Michigan, Ann Arbor, 1975.
21. K. Deb. *Multi-Objective Optimization using Evolutionary Algorithms.* John Wiley & Sons, Chichester, UK, 2001.
22. S. Droste, T. Jansen, and I. Wegener. Perhaps not a free lunch but at least a free appetizer. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 833–839. Morgan Kaufmann, 1999.
23. J. Dzubera and D. Whitley. Advanced correlation analysis of operators for the traveling salesman problem. In *Parallel Problem Solving from Nature – PPSN III*, number 866 in Lecture Notes in Computer Science, pages 68–77. Springer, 1994.
24. A.E. Eiben. Multiparent recombination. In *Evolutionary Computation 1: Basic Algorithms and Operators*, pages 289–307. Institute of Physics Publishing, 2000.
25. A.E. Eiben. Multiparent recombination in evolutionary computing. In *Advances in Evolutionary Computing*, pages 175–192. Springer, 2002.
26. A.E. Eiben, E.H.L. Aarts, and K.M. van Hee. Global convergence of genetic algorithms: A markov chain analysis. In *Parallel Problem Solving from Nature (PPSN 1)*, volume 496 of *LNCS*, pages 4–12. Springer-Verlag, 1991.
27. A.E. Eiben and T. Bäck. An empirical investigation of multiparent recombination operators in evolution strategies. *Evolutionary Computation*, 5(3):347–365, 1997.
28. A.E. Eiben and T. Bäck. Multi-parent recombination operators in continuous search spaces. Technical report, LIACS, Leiden University, 1997.
29. A.E. Eiben, R. Hinterding, and Z. Michalewicz. Parameter control in evolutionary algorithms. *IEEE Transactions on Evolutionary Computation*, 3(2):124–141, 1999.
30. A.E. Eiben, P-E. Raué, and Zs. Ruttkay. Genetic algorithms with multi-parent recombination. In *Parallel Problem Solving from Nature – PPSN III*, volume 866 of *LNCS*, pages 78–87. Springer, 1994.
31. A.E. Eiben and G. Rudolph. Theory of evolutionary algorithms: A bird's eye view. *Theoretical Computer Science*, 229(1):3–9, 1999.
32. A.E. Eiben and C.A. Schippers. Multi-parent's niche: n-ary crossover on nk-landscapes. In *Parallel Problem Solving From Nature-PPSN IV*, volume 1141 of *Lecture Notes in Computer Science*, pages 319–328. Springer-Verlag, 1996.
33. A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing.* Natural Computing. Springer-Verlag, 2003.
34. A.E. Eiben and C.H.M. van Kemenade. Performance of multi-parent crossover operators on numerical function optimization problems. Technical report, Leiden University, 1995.
35. A.E. Eiben and C.H.M. van Kemenade. Diagonal crossover in genetic algorithms for numerical optimization. *Journal of Control and Cybernetics*, 26(3):447–465, 1997.
36. A.E. Eiben, C.H.M. van Kemenade, and J.N. Kok. Orgy in the computer: Multi-parent reproduction in genetic algorithms. In *Proceedings of the Third European Conference on Artificial Life*, volume 929, pages 934–945, 1995.
37. L.J. Eshelman and J.D. Schaffer. Preventing premature convergence in genetic algorithms by preventing incest. In Rick Belew and Lashon Booker, editors, *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 115–122, San Mateo, CA, 1991. Morgan Kaufman.
38. S.C. Esquivel, H.A. Leiva, and R.H. Gallard. Multiple crossovers between multiple parents to improve search in evolutionary algorithms. In *Proceedings of the Congress on Evolutionary Computation*, volume 2, pages 1589–1594. IEEE Press, 1999.
39. S.C. Esquivel, H.A. Leiva, and R.H. Gallard. Multiplicity in genetic algorithms to face multicriteria optimization. In *Proceedings of the Congress on Evolutionary Computation*, volume 1, pages 85–90. IEEE Press, 1999.

40. D.S. Falconer and T.F.C. Mackay. *Introduction to Quantitative Genetics.* Prentice Hall, 4 edition, 1996.
41. G.B. Fogel and D.W. Corne. *Evolutionary Computation in Bioinformatics.* Morgan Kaufmann, 2002.
42. J.W. Fristrom and M.T. Clegg. *Principles of Genetics.* W.H. Freeman and Company, 1989.
43. M. Gen and R. Cheng. *Genetic Algorithms and Engineering Optimization.* Wiley Series in Engineering Design and Automation. John Wiley & Sons, 1997.
44. A. Ghosh and S. Tsutsui. *Advances in Evolutionary Computing.* Natural Computing Series. Springer, 2003.
45. F. Glover. Tabu search – part I. *ORSA Journal on Computing*, 1(3):190–206, 1989.
46. F. Glover. Tabu search – part II. *ORSA Journal on Computing*, 2(1):4–32, 1990.
47. F. Glover and M. Laguna. *Tabu Search.* Kluwer Academic Publishers, 1998.
48. D.E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning.* Addison Wesley, 1989.
49. D.E. Goldberg and R. Lingle. Alleles, loci, and the traveling salesman problem. In *Proceedings of the First International Conference on Genetic Algorithms and Their Applications*, pages 154–159. Lawrence Erlbaum Associates, Publishers, 1985.
50. D.E. Goldberg and P. Segrest. Finite markov chain analysis of genetic algorithms. In *Proceedings of the 2nd International Conference on Genetic Algorithms and their Applications*, pages 1–8. Lawrence Erlbaum Associates, 1987.
51. J. Gottlieb and G.R. Raidl, editors. *Evolutionary Computation in Combinatorial Optimization – EvoCOP 2004*, volume 3004 of *LNCS*. Springer-Verlag Inc., 2004.
52. J.J. Grefenstette and J.E. Baker. How genetic algorithms work: A critical look at implicit parallelism. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 20–27. Morgan Kaufmann Publishers, 1989.
53. D.L. Hartl and A.G. Clark. *Principles of Poppulation Genetics.* Sinauer Associates, 1 edition, 1989.
54. R. Hinterding, Z. Michalewicz, and A.E. Eiben. Adaptation in evolutionary computation: a survey. In *Proceedings of the Fourth IEEE Conference on Evolutionary Computation*, pages 65–69, 1997.
55. J. Holland. *Adaptation in Natural and Artificial Systems.* University of Michigan Press, 1975.
56. J. Horn. Finite markov chain analysis of genetic algorithms with niching. In S. Forrest, editor, *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 110–117. Morgan Kaufmann Publishers, 1993.
57. R.A. Horn and C.A. Johnson. *Matrix Analysis.* Cambridge University press, 1985.
58. L. Kallel, B. Naudts, and A. Rogers, editors. *Theoretical Aspects of Evolutionary Computing.* Springer, 2001.
59. S.A. Kauffman. *The Origins of Order: Self-Organization and Selection in Evolution.* Oxford University Press, 1993.
60. M. Kijima. *Markov Processes for Stochastic Modeling.* Chapman & Hall, 1997.
61. H. Kita, I. Ono, and S. Kobayashi. Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms. In *Proceedings of the Congress on Evolutionary Computation*, volume 2, pages 1581–1588. IEEE Press, 1999.
62. H.A. Leiva, S.C. Esquivel, and R.H. Gallard. Multiplicity and Local Search in Evolutionary Algorithms to Build the Pareto Front. In *Proceedings of the XX International Conference of the Chilean Computer Science Society*, pages 7–13. IEEE Computer Society Press, 2000.
63. J. Lis and A.E. Eiben. A multisexual genetic algorithm for multicriteria optimization. In *Proceedings of the Fourth IEEE Conference on Evolutionary Computation, Indianapolis, IN*, pages 59–64, 1997.
64. F.G. Lobo, D.E. Goldberg, and M. Pelikan. Time complexity of genetic algorithms on exponentially scaled problems. In Darrell Whitley, David Goldberg, Erick Cantu-Paz, Lee Spector, Ian Parmee, and Hans-Georg Beyer, editors, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000)*, pages 151–158. Morgan Kaufmann, 2000.

65. K. Mathias and D. Whitley. Genetic operators, the fitness landscape, and the traveling salesman problem. In *Parallel Problem Solving from Nature II.* Elsevier Science Publishers, B. V., 1992.
66. Z. Michalewicz. *Genetic Algorithms + Data Structures = Evolution Programs.* Springer-Verlag, third edition, 1996.
67. W. Millan, J. Fuller, and E. Dawson. New concepts in evolutionary search for boolean functions in cryptology. In *Proceedings of the 2003 IEEE Congress on Evolutionary Computation*, volume 3, pages 2157–2164, 2003.
68. B.L. Miller. *Noise, Sampling, and Efficient Genetic Algorithms.* PhD thesis, University of Illinois at Urbana-Champaign, Urbana, IL, 1997.
69. T.M. Mitchell. *Machine Learning.* McGraw-Hill, 1997.
70. H. Mühlenbein. Evolution in time and space – the parallel genetic algorithm. In *Foundations of Genetic Algorithms*, pages 316–337. Morgan Kaufmann, 1991.
71. H. Mühlenbein. The equation for the response to selection and its use for prediction. *Evolutionary Computation*, 5(3):303–346, 1998.
72. H. Mühlenbein and D. Schlierkamp-Voosen. Predictive models for the breeder genetic algorithm: I. continuous parameter optimization. *Evolutionary Computation*, 1(1):25–49, 1993.
73. H. Mühlenbein and D. Schlierkamp-Voosen. The science of breeding and its application to the breeder genetic algorithm (BGA). *Evolutionary Computation*, 1(4):335–360, 1994.
74. H. Mühlenbein, M. Schomisch, and J. Born. The parallel genetic algorithm as function optimizer. In *Proceedings of the 4th International Conference on Genetic Algorithms*, pages 271–278. Morgan Kaufmann, 1991.
75. T. Nagylaki. *Introduction to Theoretical Population Genetics.* Springer-Verlag, 1992.
76. H.D. Nguyen, I. Yoshihara, and M. Yasunaga. Modified edge recombination operators of genetic algorithms for the traveling salesman problem. In *Proceedings of IEEE International Conference on Industrial Electronics, Control, and Instrumentation*, pages 2815–2820, 2000.
77. A.E. Nix and M.D. Vose. Modeling genetic algorithms with markov chains. *Annals of Mathematics and Artificial Intelligence*, 5:78–88, 1992.
78. I.M. Oliver, D.J. Smith, and J.R.C. Holland. A study of permutation crossover operators on the traveling salesman problem. In *Proceedings of the 2nd International Conference on Genetic Algorithms and their Applications*, pages 224–230. Lawrence Erlbaum Associates, 1987.
79. A. Papoulis and S.U. Pillai. *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill, New York, NY, USA, 4 edition, 2002. ISBN 0-07-048477-5.
80. N.J. Radcliffe. Non-linear genetic representations. In *Parallel Problem Solving from Nature II*, pages 259–268. Elsevier Science Publishers, 1992.
81. G.R. Raidl, S. Cagnoni, J. Branke, D.W. Corne, R. Drechsler, Y. Jin, C.G. Johnson, P. Machado, E. Marchiori, F. Rothlauf, G.D. Smith, and G. Squillero, editors. *Applications of Evolutionary Computing: EvoWorkshops 2004: EvoBIO, EvoCOMNET, EvoHOT, EvoIASP, EvoMUSART, and EvoSTOC*, volume 3005 of *Lecture Notes in Computer Science.* Springer-Verlag, 2004.
82. G.R. Raidl and J. Gottlieb, editors. *Evolutionary Computation in Combinatorial Optimization – EvoCOP 2005*, volume 3448 of *LNCS.* Springer Verlag, 2005.
83. A. Rogers and A. Prügel-Bennett. Genetic drift in genetic algorithm selection schemes. *IEEE Transactions on Evolutionary Computation*, 3(4):298–303, 1999.
84. F. Rothlauf, J. Branke, S. Cagnoni, D.W. Corne, R. Drechsler, Y. Jin, P. Machado, E. Marchiori, J. Romero, G.D. Smith, and G. Squillero, editors. *Applications on Evolutionary Computing: EvoWorkshops 2005: EvoBIO, EvoCOMNET, EvoHOT, EvoIASP, EvoMUSART, and EvoSTOC*, volume 3449 of *Lecture Notes in Computer Science.* Springer-Verlag, 2005.
85. G. Rudolph. Convergence analysis of canonical genetic algorithms. *IEEE Transactions on Neural Networks*, 5(1):96–101, 1994.
86. G. Rudolph. Finite Markov chain results in evolutionary computation: A tour d'horizon. *Fundamenta Informaticae*, 35:67–89, 1998.
87. R. Salomon. Reevaluating genetic algorithm performance under coordinate rotation of benchmark functions: A survey of some theoretical and practical aspects of genetic algorithms. *BioSystems*, 39:263–278, 1996.

88. R. Salomon. Raising theoretical questions about the utility of genetic algorithms. In *Evolutionary Programming VI*, volume 1213 of *Lecture Notes in Computer Science*, pages 275–284. Springer, 1997.
89. K. Sastry and D.E. Goldberg. Let's get ready to rumble: Crossover versus mutation head to head. In *Genetic and Evolutionary Computation – GECCO-2004, Part II*, volume 3103 of *Lecture Notes in Computer Science*, pages 126–137. Springer-Verlag, 2004.
90. C.A. Schippers. Multi-parent scanning crossover and genetic drift. In *Theoretical Aspects of Evolutionary Computing*, pages 307–330. Springer, 2001.
91. L.M. Schmitt. Theory of genetic algorithms. *Theoretical Computer Science*, 259(1–2):1–61, 2001.
92. L.M. Schmitt. Theory of Genetic Algorithms II: Models for genetic operators over the string-tensor representation of populations and convergence to global optima for arbitrary fitness function under scaling. *Theoretical Computer Science*, 310(1–3):181–231, 2004.
93. H.-P. Schwefel. *Evolution and Optimum Seeking.* John Wiley & Sons, Inc., 1995.
94. H. Shimodaira. Dcga: a diversity control oriented genetic algorithm. In *Proceedings of the Second International Conference on Genetic Algorithms In Engineering Systems:Innovations And Applications*, pages 444–449, 1997.
95. W.M. Spears. *The Role of Mutation and Recombination in Evolutionary Algorithms.* PhD thesis, George Mason University, 1998.
96. I.G. Sprinkhuizen-Kuyper, C.A. Schippers, and A.E. Eiben. On the real arity of multiparent recombination. In *Proceedings of the Congress on Evolutionary Computation*, volume 1, pages 680–686. IEEE Press, 1999.
97. M. Srinivas and L.M. Patnaik. Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions on Systems, Man and Cybernetics*, 24(4):656–667, 1994.
98. T. Starkweather, S. McDaniel, K. Mathias, D. Whitley, and C. Whitley. A comparison of genetic sequencing operators. In *Proceedings of the Fourth International Conference on Genetic Algorithms*, pages 69–76. Morgan Kaufman, 1991.
99. G. Syswerda. Uniform crossover in genetic algorithms. In *Proceedings of the 3rd International Conference on Genetic Algorithms*, pages 2–9. Morgan Kaufmann, 1989.
100. D. Thierens, D.E. Goldberg, and Â.G. Pereira. Domino convergence, drift, and the temporal-salience structure of problems. In *Proceedings of 1998 IEEE Iternational Conference on Evolutionary Computation*, pages 535–540, 1998.
101. C.K. Ting. An analysis of the effectiveness of multi-parent crossover. In *Parallel Problem Solving from Nature VIII*, volume 3242 of *LNCS*, pages 131–140. Springer-Verlag, 2004.
102. C.K. Ting. Improving edge recombination through alternate inheritance and greedy manner. In *Evolutionary Computation in Combinatorial Optimization – EvoCOP 2004*, volume 3004 of *LNCS*, pages 210–219. Springer-Verlag, 2004.
103. C.K. Ting. On the convergence of multi-parent genetic algorithms. In *Proceedings of the 2005 Congress on Evolutionary Computation.* IEEE Press, 2005.
104. C.K. Ting. On the mean convergence time of multi-parent genetic algorithms without selection. In *Proceedings of the Eighth European Conference on Artificial Life*, volume 3630 of *LNAI*, pages 403–412. Springer-Verlag, 2005.
105. C.K. Ting and H. Kleine Büning. A mating strategy for multi-parent genetic algorithms by integrating tabu search. In *Proceedings of the 2003 Congress on Evolutionary Computation*, pages 1259–1266. IEEE Press, 2003.
106. C.K. Ting, S.T. Li, and C.N. Lee. Tga: A new integrated approach to evolutionary algorithms. In *Proceedings of the 2001 Congress on Evolutionary Computation*, pages 917–924. IEEE Press, 2001.
107. C.K. Ting, S.T. Li, and C.N. Lee. On the harmonious mating strategy through tabu search. *Information Sciences*, 156:189–214, 2003.
108. S. Tsutsui. Multi-parent recombination in genetic algorithms with search space boundary extension by mirroring. In *Parallel Problem Solving from Nature – PPSN V*, volume 1498 of *Lecture Notes in Computer Science*, pages 428–437. Springer, 1998.

109. S. Tsutsui and A. Ghosh. A study on the effect of multi-parent recombination in real coded genetic algorithms. In *Proceedings of International Conference on Evolutionary Computation*, pages 828–833, 1998.
110. S. Tsutsui and L.C. Jain. On the effect on multi-parent recombination in real coded genetic algorithms. In *Proceedings of the 2nd International Conference on Knowledge-based Intelligent electronic systems*, pages 155–160, 1998.
111. S. Tsutsui, M. Yamamura, and T. Higuchi. Multi-parent recombination with simplex crossover in real coded genetic algorithms. In *Proceedings of the Genetic and Evolutionary Computation Conference*, volume 1, pages 657–664. Morgan Kaufmann, 1999.
112. H. Voigt and H. Mühlenbein. Gene pool recombination and the utilization of covariances for the breeder genetic algorithm. In *Proceedings of the 2nd IEEE International Conference on Evolutionary Computation*, pages 172–177. IEEE Press, 1995.
113. M.D. Vose. *The Simple Genetic Algorithm: Foundations and Theory.* MIT Press, Cambridge, MA, 1999.
114. D. Whitley. The genitor algorithm and selection pressure: Why rank-based allocation of reproductive trials is best. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 116–121. Morgan Kaufmann Publishers, 1989.
115. D. Whitley, T. Starkweather, and D. Fuquay. Scheduling problems and traveling salesman: The genetic edge recombination operator. In *Proceedings of the Third International Conference on Genetic Algorithms*, pages 133–140. Morgan Kaufman, 1989.
116. D. Whitley, T. Starkweather, and D. Shaner. The traveling salesman and sequence scheduling: Quality solutions using genetic edge recombination. In *Handbook of Genetic Algorithms*, pages 350–372. Van Nostrand Reinhold, 1991.
117. D.H. Wolpert and W.G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.
118. E. Zitzler and L. Thiele. Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *IEEE Transactions on Evolutionary Computation*, 3(4):257–271, 1999.