A Fixed Point Analysis of a Gene Pool GA with Mutation

Alden H. Wright *
Computer Science
University of Montana
USA
wright@cs.umt.edu
44 121 414 2793

Jonathan E. Rowe
Computer Science
University of Birmingham
UK
j.e.rowe@cs.bham.ac.uk

Riccardo Poli
Computer Science
University of Essex
UK
rpoli@essex.ac.uk

Christopher R. Stephens
Instituto de Ciencias Nucleares
UNAM, Mexico
stephens@nuclecu.unam.mx

Abstract

This paper analyzes a recombination/mutation/selection genetic algorithm that uses gene pool recombination. For linear fitness functions, the infinite population model can be described by ℓ equations where ℓ is the string length. For linear fitness functions, we show that there is a single fixed point and that this fixed point is stable. For the ONEMAX fitness function, the model reduces to a linear recurrence in a single variable which can be explicitly solved. The time-to-convergence for ONEMAX is given.

1 Introduction

A major goal of genetic algorithm theory is a tractable model of a GA that gives quantitative predictions over multiple generations. The Vose dynamical system model [9] is exact in the infinite-population limit, but it tends to be intractable for results on specific problems due to the fact that the model keeps track of the frequency of every string. What is needed is a "coarse-grained" model that simplifies the model. One natural way to attempt to coarse-grain a GA model is to look at the representations of schemata, especially low-order schemata. (See [7] and [10].) One would like to track schema averages over multiple generations.

Crossover does not change the schema frequencies of order 1 schemata. Stated another way, it preserves the frequencies of each allele. The effect of crossover on a population is to move the population closer to linkage equilibrium. In other words, it decorrelates the alleles at different positions. In a linkage equilibrium population, the representation of any string is determined by the allele frequencies. Geiringer's theorem [1] shows that the limit of repeated applications of crossover is a population in linkage equilibrium (also known as Robbins' proportions).

In gene pool recombination, the population is taken directly to linkage equilibrium in one step. In other words, the alleles are completely decorrelated, and the population is completely described by the allele frequencies. This can be implemented for a finite population by choosing the genes for a new individual from a pool constructed from the whole population. In other words, the population allele distribution at a particular locus defines a probability distribution, and the allele at that position in a new individual is selected from that probability distribution. Gene pool recombination was introduced by Voigt and Mühlenbein in [8].

After a gene pool recombination step, the complete state of the population is determined by the allele frequencies, and the number of these variables is linear rather than exponential in the string length. This "coarse-graining" makes the gene pool much more tractable for analysis than the two-parent recombination GA, and this is the focus of this paper.

Mühlenbein, Mahnig and others have done considerable work on the Univariate Marginal Distribution Algorithm (UMDA) [6], [4]. This algorithm uses gene pool recombination, and selection, and no mutation. Response to selection is computed or approximated for a variety of selection methods. In [3], Mahnig and Mühlenbein introduce mutation into UMDA by a technique called Bayesian prior. The emphasis is on the interaction of the mutation rate and the population size.

By using the Walsh basis, we are able to analyze a GA that uses gene pool recombination, selection, and mutation. We find and rigorously prove the stability of fixed points which is a different approach than has been taken in the work on UDMA which is primarily oriented towards "response to selection". Our work shows how the Vose dynamical system model framework can be used to model algorithms like UMDA.

We will call the GA that uses gene pool recombination,

---

[*] This paper was written while Alden Wright was visiting the School of Computer Science, University of Birmingham, UK, supported by EPSRC grant GR/R47394.

selection, and mutation a gene pool GA.

The paper considers linear fitness functions. Other classes of fitness functions, such as the class of single-peak (needle-in-the-haystack) fitness functions will be considered elsewhere.

There has been a long history of research on the analysis of fitness functions by means of their Walsh coefficients. The Vose model [9] uses the Walsh transform to simplify the model for crossover and mutation. This paper makes a connection between these two lines of research. It should be the first step in a research program that uses the Walsh transform to analyze selection/mutation/recombination genetic algorithms.

In addition, the gene pool GA can suggest general properties of recombination that can then be tested for a GA that uses two-parent recombination.

We show that for any linear fitness function over fixed length binary strings, there is a unique fixed point which is asymptotically stable. For the ones-counting (ONEMAX) fitness function, explicit formulas are given for all fixed points of the $\mathcal{G}$ function that defines the infinite population model. To our knowledge, this is the first time that formulas for fixed points have been given for a GA that involves selection, recombination, and mutation and for arbitrary string length.

A gene pool recombination GA can either be used as an approximation to a two-parent recombination GA or as an alternative GA. We show empirically that the gene pool GA is a good approximation to a two-parent recombination GA for linear fitness functions.

For all results where separate proofs are given, the proofs are in the appendix.

2 Notation

Notation in this paper mostly follows [9].

The search space for this paper is the set of all binary strings of length ℓ which will be denoted Ω. The binary representation of a string induces a correspondence from the elements of Ω to the set of integers from 0 to $2^\ell - 1$. Thus, the integer 0 corresponds to the all-zeros string, and the integer $2^\ell - 1$ corresponds to the all-ones string. A sum over $i \in \Omega$ is equivalent to a sum from $i = 0$ to $i = 2^\ell - 1$.

The bitwise mod 2 sum of two strings j and k is denoted by $j \oplus k$; Ω is a group under this operation. Note that $j \oplus k$ is also the XOR of j and k. The identity element is the string of zeros, which will be denoted by 0. The bitwise product of strings j and k is denoted by $j \otimes k$. The ones-complement of k is denoted by $\overline{k}$.

The number of ones in a binary string k is denoted by $\#k$.

For each $u \in \Omega$, $\Omega_u = \{i \in \Omega : i \otimes u = i\}$; Ω_u is the set of binary strings which have a 1 only in positions where u has a 1. It is also a schema denoted by a string over $\{0,*\}$ where there are asterisks in those positions where the corresponding bit of u is 1, and where the fixed positions are all zeros. For example, if $\ell = 4$ and $u = 0101$, then Ω_u is the special schema $0{*}0{*}$, and is also the set $\{0000, 0001, 0100, 0101\}$. Note that Ω_u is a subgroup of Ω.

Let $\mathcal{L} = \{j \in \Omega : \#j = 1\}$. Under the identification of Ω with the integers, $\mathcal{L} = \{2^k : k = 0, 1, \ldots, \ell - 1\}$. Let $\mathcal{L}_u = \{j \in \mathcal{L} : j \otimes u = j\} = \mathcal{L} \cap \Omega_u$. For example, if $\ell = 6$ and $u = 21$, $\mathcal{L}_u = \{2^0, 2^2, 2^4\} = \{1, 4, 16\} = \{000001, 000100, 001000\}$.

The set $\mathcal{L}$ will be used extensively as an index set. This might seem unnatural since $\mathcal{L}$ is a subset of Ω which does not correspond to consecutive integers. If this bothers the reader, a product over $i \in \mathcal{L}$, such as $\prod_{i \in \mathcal{L}} S_i$, could be rewritten as $\prod_{k=0}^{\ell-1} S_{2^k}$.

A population (a multiset of Ω) is represented as a population vector x indexed by Ω; $x_v = x_{v_0 v_1 \ldots v_{\ell-1}}$ is the fraction of the population which is string $v = v_0 v_1 \ldots v_{\ell-1}$, where $v_0, v_1, \ldots, v_{\ell-}$ are the bits of v. For example, if $\ell = 2$ and the population as a multiset is $\{00, 01, 01, 11\}$, then the corresponding population vector is $[\frac{1}{4},\ \frac{1}{2},\ 0,\ \frac{1}{4}]^T$. A population vector is a population-size independent representation of a population, and thus is natural for infinite-population models.

If a population x depends on time (or GA generation) t, then $x(t)$ is the population at time t.

All population vectors are contained in the simplex $\Lambda = \{x : \sum_{j \in \Omega} x_j = 1 \text{ and } x_j \geq 0 \text{ for all } j\}$. The simplex is a natural setting for the dynamical systems model since it allows population vectors to range continuously over a subset of $\mathbb{R}^{2^\ell}$ and thus allows derivatives and calculus to be used.

The *Walsh matrix* W is an 2^ℓ by 2^ℓ matrix defined by $W_{i,j} = 2^{-\ell/2}(-1)^{\#(i \otimes j)}$ The Walsh matrix is symmetric and $W = W^{-1}$. For example, the Walsh matrix for $\ell = 2$ is:

$$W = \frac{1}{2}\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

If x is a population vector, then the Walsh transform of x is Wx and is denoted by $\widehat{x}$. Let $e_0, e_1, \ldots, e_{N-1}$ be the standard basis vectors for $\mathbb{R}^N$. Then the vectors $\widehat{e}_0, \widehat{e}_1, \ldots, \widehat{e}_{N-1}$ form the Walsh basis for $\mathbb{R}^N$. If x is a vector then $\widehat{x} = Wx$ is expressed in the Walsh basis. In other words, x_j is the jth coordinate of x in the standard basis and $\widehat{x}_j$ is the jth coordinate of x in the Walsh basis. If A is a 2^ℓ by 2^ℓ matrix, then WAW is the Walsh transform of A and is denoted by $\widehat{A}$.

3 Linkage Equilibrium

This section gives some basic results relating schema averages, the Walsh representation of a population, and properties of a population that is at linkage equilibrium.

These results will show how string and schema frequencies can be determined from the Walsh coefficients that we use to describe our models.

If x is a population vector and $u \in \mathcal{L}$, let

$$x_v^{(u)} = \sum_{i \in \Omega} x_i \delta_{i \otimes u, v}$$

where $\delta_{j,k} = 1$ iff $j = k$. In other words, $x_v^{(u)}$ denotes the proportion of individuals i in the population such that the position of i corresponding to u is the same as the corresponding position of v. For example, if $x = \frac{1}{64}[19,\ 5,\ 29,\ 11]^T$, then $x_{00}^{(01)} = 48/64$ and $x_{01}^{(01)} = 16/64$.

Definition 1 *A population x is in* **linkage equilibrium** *if*

$$x_k = \prod_{u \in \mathcal{L}} x_{u \otimes k}^{(u)}$$

Thus, a population is in linkage equilibrium if the frequency of each string is the product of the corresponding 1 schema averages. For example, the population $\frac{1}{64}[19,\ 5,\ 29,\ 11]^T$ (with Walsh basis representation $\frac{1}{32}[16,\ 8,\ -4,\ -1]^T$) is in linkage equilibrium.

The following relates the schema sums of the order 1 schemata and the order 1 Walsh coefficients. It follows from the formula for the Walsh coefficients.

Lemma 2 *If $u \in \mathcal{L}$,*

$$x_v^{(u)} = \frac{1}{2}\left(1 + (-1)^{\#v} 2^{\ell/2} \widehat{x}_u\right)$$

The frequency of any string can be found from the Walsh basis representation of the population.

Corollary 3 *Let x be in linkage equilibrium. For any $v \in \Omega$,*

$$x_v = 2^{-\ell} \prod_{i \in \mathcal{L}} \left(1 + (-1)^{\#(i \otimes v)} 2^{\ell/2} \widehat{x}_i\right)$$

The following theorem gives the relationship of linkage equilibrium to the Walsh basis representation of a population. The result follows easily from theorem 10.9 of [9].

Theorem 4 *If population x is in linkage equilibrium, then*

$$\widehat{x}_k = 2^{(\#k-1)\ell/2} \prod_{i \in \mathcal{L}_k} \widehat{x}_i$$

Remark: While higher-order schemata will not be used in this paper, it is interesting to note that the higher order schemata are products of order 1 schemata for a population at linkage equilibrium. Thus, in traditional notation, the frequency of the schema $0{*}1{*}$ is the product of the frequencies of $0{*}{*}{*}$ and ${*}{*}1{*}$.

Theorem 5 *If $x \in \Lambda$, then $|\widehat{x}_k| \leq 2^{-\ell/2}$ for all $k \in \Omega$.*

4 The Gene Pool Model in the Walsh Basis

In this paper, one generation of the gene pool GA will consist of the following three steps:

1. Gene pool recombination.
2. Proportional selection.
3. Mutation.

The dynamical system model will be described by a function $\mathcal{G} : \Lambda \longrightarrow \Lambda$. In other words, if $x(t)$ is the population at generation t, then the population at generation $t+1$ is given by $x(t+1) = \mathcal{G}(x(t))$.

$\mathcal{G}$ is a composition of $\mathcal{M}$, $\mathcal{F}$, and $\mathcal{U}$ (i. e., $\mathcal{G} = \mathcal{U} \circ \mathcal{F} \circ \mathcal{M}$). The $\mathcal{M}$, $\mathcal{F}$, and $\mathcal{U}$ functions are described below.

The next three subsections tell how to compute $\mathcal{F}$, $\mathcal{U}$, and $\mathcal{M}$ in the Walsh basis.

4.1 Proportional selection

Following [9], the effect of proportional selection can be described by a function $\mathcal{F} : \Lambda \longrightarrow \Lambda$. The probability that an individual $k \in \Omega$ is chosen to be in the new population is $\mathcal{F}(x)_k$. Or stated another way, if proportional selection is applied to a population x, then the expected frequency vector for the resulting population is $\mathcal{F}(x)$.

If $f \in \mathbb{R}^{2^\ell}$ is the fitness function (i. e., f_k is the fitness of k) then

$$\mathcal{F}(x) = \frac{Fx}{f^T x}$$

where F is the diagonal matrix such that $F_{k,k} = f_k$.

The average fitness $f^T x$ when computed in the Walsh basis is the same: $\widehat{f^T}\widehat{x} = (f^T W)(Wx) = f^T x$. Thus,

$$\widehat{\mathcal{F}(x)} = \frac{WFx}{\widehat{f}^T \widehat{x}} = \frac{WFWWx}{\widehat{f}^T \widehat{x}} = \frac{\widehat{F}\widehat{x}}{\widehat{f}^T \widehat{x}}$$

Lemma 6 *For any $i, j \in \Omega$,*

$$\widehat{F}_{i,j} = 2^{-\ell/2} \widehat{f}_{i \oplus j}$$

Let σ_k denote the $2^\ell \times 2^\ell$ matrix defined by $(\sigma_k)_{i,j} = \delta_{i\oplus k,j}$. Then it is easy to show that $(\sigma_k x)_i = x_{i\oplus k}$.

The following corollary tells how to compute proportional selection in the Walsh basis.

Corollary 7

$$\widehat{F(x)}_k = \frac{\sum_{i\in\Omega}\widehat{f}_{i\oplus k}\widehat{x}_i}{2^{\ell/2}\widehat{f}^T\widehat{x}} = \frac{\sum_{i\in\Omega}\widehat{f}_i\widehat{x}_{i\oplus k}}{2^{\ell/2}\widehat{f}^T\widehat{x}} = \frac{\widehat{f}^T\sigma_k\widehat{x}}{2^{\ell/2}\widehat{f}^T\widehat{x}}$$

4.2 Mutation

The effect of mutation can be described by a function $\mathcal{U} : \Lambda \longrightarrow \Lambda$. The probability that an individual $k \in \Omega$ is chosen to be in the new population is $\mathcal{U}(x)_k$.

If the probability that a bit is mutated (flipped) is μ, then the probability that an individual $i \in \Omega$ is mutated to an individual j is $\mu^{\#(i\oplus j)}(1-\mu)^{\ell-\#(i\oplus j)}$. (Note that $\#(i\oplus j)$ is the Hamming distance from i to j.) Thus, we can define a $2^\ell \times 2^\ell$ mutation matrix U with $U_{i,j} = \mu^{\#(i\oplus j)}(1-\mu)^{\ell-\#(i\oplus j)}$, and $\mathcal{U}(x) = Ux$.

Following Vose (section 4.3 of [9]), we define a vector μ indexed over Ω by

$$\mu_i = (\mu)^{\#i}(1-\mu)^{\ell-\#i}$$

(Whether μ denotes the scalar mutation rate or the mutation vector should be clear from the context.) The following lemma is proved in [9].

Lemma 8
$$\widehat{\mu}_k = 2^{-\ell/2}(1-2\mu)^{\#k}$$

The Walsh transform of U is easily shown to be diagonal.

Lemma 9 *$\widehat{U}$ is a diagonal matrix which diagonal entries given by:*
$$\widehat{U}_{i,i} = (1-2\mu)^{\#i}$$

Theorem 10

$$\widehat{\mathcal{U}(x)}_k = (\widehat{U}\widehat{x})_k = (1-2\mu)^{\#k}\widehat{x}_k$$

Thus, mutation is very simple in the Walsh basis: it corresponds to multiplication by a diagonal matrix.

4.3 Gene pool recombination

To implement gene pool recombination in a finite population GA, the order 1 schema sums $x^{(u)}$ for $\#u = 1$ are computed from the current population x. Then each individual of the new population is constructed by choosing each bit according the probabilities determined by these schema sums. Thus, the probability that the bit at position i is 0 is $x_0^{(2^i)}$ and the probability that the bit at position i is 1 is $x_{2^i}^{(2^i)}$.

In the infinite population model, the population y resulting from applying gene pool recombination to population x has the property that $\widehat{y}_u = \widehat{x}_u$ for $\#u = 1$. In addition, y is at linkage equilibrium. Thus, the Walsh coefficients of the population after gene pool recombination can be computed from theorem 4.

As with proportional selection and mutation, we can describe gene pool recombination through a function $\mathcal{M} : \Lambda \longrightarrow \Lambda$. By theorem 4, we have:

$$\widehat{\mathcal{M}(x)}_k = 2^{(\#k-1)\ell/2}\prod_{i\in\mathcal{L}_k}\widehat{x}_i = 2^{-\ell/2}\prod_{i\in\mathcal{L}_k}2^{\ell/2}\widehat{x}_i$$

This formula tells how to compute gene pool recombination in the Walsh basis. Note that if $k \in \mathcal{L}$ (i. e., $\#k = 1$), then $\widehat{M(x)}_k = \widehat{x}_k$. In other words, $\mathcal{M}$ is the identity on the order-1 Walsh coefficients.

5 Linear fitness and the ONEMAX problem

Definition 11 *A fitness function represented by a fitness vector f is* **linear** *if*

$$f_i = c + \sum_{j\in\mathcal{L}} b_j\delta_{i\otimes j,j}$$

where c and b_j, $j \in \mathcal{L}$ are constants. (Note that if $j = 2^k \in \mathcal{L}$, then $i \otimes j = j$ if and only if bit k of i is 1.)

Without loss of generality, we can assume that $b_j \geq 0$ for $j \in \mathcal{L}$. If some $b_j < 0$ where $j = 2^p$, this says that a string with 0 at position p is more fit than the same string with a 1 at that position. Thus, a change of representation where 0 and 1 are interchanged in all strings at that position will make $b_j > 0$.

The ONEMAX fitness function is the linear fitness function where $c = 0$ and $b_j = 1$ for all $j \in \mathcal{L}$.

The following lemma shows that when a linear fitness is expressed in the Walsh basis, only the order 0 and order 1 coefficients are nonzero.

Lemma 12 *For any $k \in \Omega$,*

$$\widehat{f}_k = \begin{cases} 2^{\ell/2}c + 2^{\ell/2-1}\sum_{j\in\mathcal{L}} b_j & \text{if } k = 0 \\ -2^{\ell/2-1}b_k & \text{if } k \in \mathcal{L} \\ 0 & \text{otherwise} \end{cases}$$

If $c \geq 0$ and $b_j \geq 0$ for $j \in \mathcal{L}$, then $\widehat{f}_j \leq 0$ and $\widehat{f}_0 \geq -\sum_{j\in\mathcal{L}}\widehat{f}_j$.

Since we are working in the Walsh basis, it will convenient to simply characterize a linear fitness in terms of its Walsh coefficients. So in this section, we assume that $\widehat{f}_k = 0$ for $k \notin \mathcal{L} \cup \{0\}$, $\widehat{f}_j \leq 0$ for $j \in \mathcal{L}$ and $\widehat{f}_0 + \sum_{j\in\mathcal{L}}\widehat{f}_j \geq 0$.

These correspond to the assumptions that fitness is linear, $b_j \geq 0$, and $c \geq 0$.

We can now compute selection in the Walsh basis. For $k \in \mathcal{L}$,

$$\begin{aligned}\widehat{\mathcal{F}(y)}_k &= \frac{\widehat{f}^T \sigma_k \widehat{y}}{2^{\ell/2}\widehat{f}^T \widehat{y}} \\ &= \frac{\sum_{i\in\Omega} \widehat{f}_i \widehat{y}_{i\oplus k}}{2^{\ell/2}\sum_{i\in\Omega} \widehat{f}_i \widehat{x}_i} \\ &= \frac{2^{-\ell/2}\widehat{f}_k + \widehat{f}_0 \widehat{y}_k + 2^{\ell/2}\sum_{j\in\mathcal{L}\setminus\{k\}} \widehat{f}_j \widehat{y}_{j\oplus k}}{\widehat{f}_0 + 2^{\ell/2}\sum_{j\in\mathcal{L}} \widehat{f}_j \widehat{y}_j}\end{aligned}$$

Now assume that $y = \mathcal{M}(x)$ where $\mathcal{M}$ denotes gene pool recombination. Then $\widehat{y}_k = \widehat{x}_k$ and $\widehat{y}_{j\oplus k} = 2^{-\ell/2}\widehat{x}_j\widehat{x}_k$ for for $j, k \in \mathcal{L}$ by theorem 4.

For $k \in \mathcal{L}$,

$$\begin{aligned}\widehat{\mathcal{G}(x)}_k &= \widehat{e}_k^T \mathcal{U}(\mathcal{F}(\mathcal{M}(x)))_k \\ &= (1-2\mu)\frac{2^{-\ell/2}\widehat{f}_k + \widehat{f}_0 \widehat{x}_k + 2^{\ell/2}\widehat{x}_k \sum_{j\in\mathcal{L}\setminus\{k\}} \widehat{f}_j \widehat{x}_j}{\widehat{f}_0 + 2^{\ell/2}\sum_{j\in\mathcal{L}} \widehat{f}_j \widehat{x}_j}\end{aligned}$$

This formula is a recurrence that defines the gene pool model for linear fitness in terms of the variables $\widehat{x}_k$ for $k \in \mathcal{L}$.

Notation is simplified by a variable substitution. Let $\widehat{z}_k = 2^{\ell/2}\widehat{x}_k$. Then if z is in the simplex, lemma 5 shows that $-1 \leq \widehat{z} \leq 1$. The recurrence written in terms of the $\widehat{z}_k$ is:

$$\begin{aligned}\widehat{\mathcal{G}(z)}_k &= (1-2\mu)\frac{\widehat{f}_k + \widehat{f}_0 \widehat{z}_k + \widehat{z}_k \sum_{j\in\mathcal{L}\setminus\{k\}} \widehat{f}_j \widehat{z}_j}{\widehat{f}_0 + \sum_{j\in\mathcal{L}} \widehat{f}_j \widehat{z}_j} \\ &= (1-2\mu)\widehat{z}_k + (1-2\mu)\frac{\widehat{f}_k(1-\widehat{z}_k^2)}{\widehat{f}_0 + \sum_{j\in\mathcal{L}} \widehat{f}_j \widehat{z}_j} \qquad (1)\end{aligned}$$

Later we will want to be able to evaluate the average fitness in terms of the $\widehat{z}_j$. The average fitness is

$$\begin{aligned}f^T x = \widehat{f}^T \widehat{x} &= 2^{-\ell/2}\widehat{f}_0 + \sum \widehat{f}_j \widehat{x}_j \\ &= 2^{-\ell/2}\widehat{f}_0 + \sum \widehat{f}_j 2^{-\ell/2}\widehat{z}_j \\ &= 2^{-\ell/2}(\widehat{f}_0 + \sum \widehat{f}_j \widehat{z}_j)\end{aligned}$$

The fixed point equations are:

$$\widehat{f}_k \widehat{z}_k^2 + \frac{2\mu}{1-2\mu}\left(\widehat{f}_0 + \sum_{j\in\mathcal{L}} \widehat{f}_j \widehat{z}_j\right)\widehat{z}_k - \widehat{f}_k = 0 \qquad (2)$$

Lemma 13 *If fitness is linear and $0 < \mu < 1/2$, then the gene pool GA has a unique fixed point in the simplex.*

Theorem 14 *For linear fitness and $0 < \mu < 1/2$, the gene pool GA model has a unique fixed point, and this fixed point is asymptotically stable.*

5.1 The ONEMAX problem

For a rescaled ONEMAX fitness function, we can take $\widehat{f}_0 = \ell$ and $\widehat{f}_j = -1$ for all $j \in \mathcal{L}$. Rescaling has no effect on the $\mathcal{G}$ function or the fixed points, but it does affect the total fitness. From lemma 12, this original ONEMAX fitness is $2^{\ell/2-1}$ times the rescaled ONEMAX fitness. From an earlier remark, the average rescaled fitness is $2^{-\ell/2}(\widehat{f}_0 + \sum \widehat{f}_j \widehat{z}_j) = 2^{-\ell/2}(\ell - \sum \widehat{z}_j)$. Thus, the average ONEMAX fitness is $\frac{1}{2}(\widehat{f}_0 + \sum \widehat{f}_j \widehat{z}_j) = \frac{1}{2}(\ell - \sum \widehat{z}_j)$.

If we assume a symmetric population, i. e., if we assume that $\widehat{z}_j = w$ for all j, then the recurrence simplifies to:

$$\begin{aligned}\mathcal{G}(w(t)) = w(t+1) &= (1-2\mu)\frac{(\ell-1)w(t) - 1}{\ell} \\ &= \frac{(1-2\mu)(\ell-1)}{\ell}w(t) - \frac{1-2\mu}{\ell}\end{aligned}$$

For $\mu = 0$, this recurrence is equivalent to one given in [4].

This linear recurrence can easily be solved. Let $A = \frac{(1-2\mu)(\ell-1)}{\ell}$, $C = \frac{1-2\mu}{\ell}$.

$$\mathcal{G}^t(w) = w(t) = A^t w(0) - C\left(\frac{A^t - 1}{A - 1}\right)$$

The fixed point is:

$$w_{fixed} = \frac{-C}{A-1} = -\frac{1-2\mu}{2\mu(\ell-1)+1}$$

We can evaluate the average fitness for the original ONEMAX fitness at this fixed point. From above, the average ONEMAX fitness is $\frac{1}{2}(\ell - \sum \widehat{z}_j) = \frac{1}{2}(\ell - \ell w) = \frac{\ell}{2}(1-w)$. At the fixed point $w = w_{fixed}$, this evaluates to $\frac{\ell(1+\mu\ell-2\mu)}{1+2\mu\ell-2\mu}$. If the mutation rate is $\mu = \frac{1}{\ell}$, this is $\frac{\ell(2-2/\ell)}{3-2/\ell} \approx \frac{2\ell}{3}$.

Theorem 15 *Let $\epsilon > 0$ and let $\alpha = \ell\mu$ so that $\mu = \alpha/\ell$. If the gene pool GA model is started on the ONEMAX fitness function from a random population ($w(0) = 0$), and if t generations are done where t is chosen so that*

$$t > \frac{-\ell \ln \epsilon}{1+2\alpha}$$

then

$$\frac{|\mathcal{G}^t(0) - w_{fixed}|}{|w_{fixed}|} < \epsilon$$

In other words, the relative error after t generations is at most ϵ.

The theorem says that for a fixed mutation rate, the number of generations is $\Theta(\ell)$. Section 13.2 of [9] says that $\mathcal{G}$ for general fitness is logarithmically convergent. Theorem 15 agrees with this result, only it gives explicit values for the constants for the ONEMAX fitness function.

The theorem also says that the time to convergence goes down as the mutation rate increases. However, as the mutation rate increases, the fixed point moves closer to the middle of the simplex, so the algorithm has less far to go. When the mutation rate is $1/2$, the fixed point is at the center of the simplex, and the algorithm starts at the fixed point.

5.2 Empirical comparisons

The gene pool GA gives a good approximation to a two-parent GA for linear fitness functions. Figure 5.2 show the average fitness and figure 5.2 shows the number of optimal individuals for 99 generations with gene pool, uniform, and one-point recombination, where a crossover rate of 1 is used for uniform and one-point recombination. The string length was 18, and the mutation rate was 0.01. The graph shows an average of 10 runs, each with a population size of 500000. The errors are negligible.

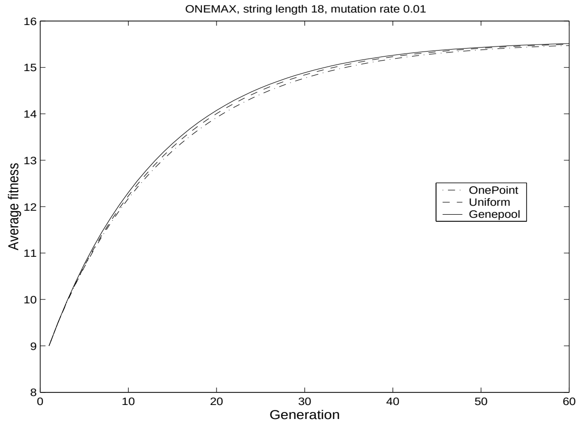

Figure 1: The average fitness for different types of recombination

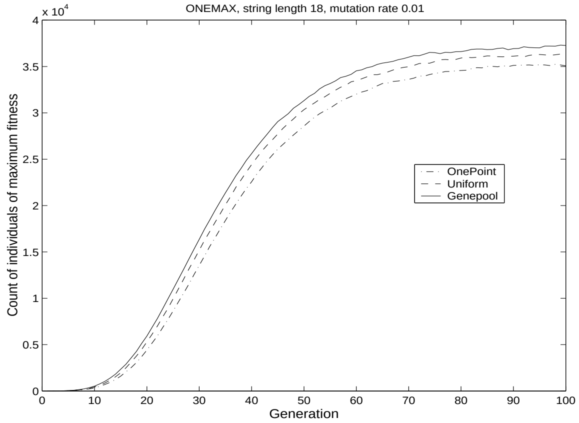

Figure 2: The number of optimal individuals for different types of recombination

6 Discussion and Conclusion

In the paper we have given an exact infinite population model of a selection/mutation/recombination genetic algorithm that is tractable for large string lengths. We have shown that for linear fitness functions, there is a unique fixed point, and this fixed point is asymptotically stable in the space of all populations. For the ONEMAX problem, the model reduces to a single variable, and an explicit solution to the recurrence equation is given for symmetric populations. The fixed point equation is a single quadratic equation in one variable, so an explicit formula for the fixed point is given. The time to convergence is shown to be $O(\ell)$ generations, where ℓ is the string length.

This paper should be the first step in the unification of the Walsh basis analysis of crossover/mutation given in [9] with the Walsh basis analysis of fitness functions.

7 Appendix

First is a technical lemma.

Lemma 16

$$\sum_{j\in\Omega_u}(-1)^{\#(j\otimes w)} = \begin{cases} 2^{\#u} & \text{\textit{if} } w\in\Omega_{\overline{u}} \\ 0 & \text{\textit{otherwise}} \end{cases}$$

Proof. Suppose that $w \in \Omega_{\overline{u}}$. This implies that $j \otimes w = 0$ for all $j \in \Omega_u$. Thus, the summation is equal to the number of elements in Ω_u, which is $2^{\#u}$.

Now suppose $w \notin \Omega_{\overline{u}}$. The we can write $w = q \oplus v$ where $\#q = 1$, $q \in \Omega_u$, and $q \otimes v = 0$. Note that $\Omega_q = \Omega_{u\otimes q} = \{0, q\}$. Then

$$\sum_{j\in\Omega_u}(-1)^{\#(j\otimes w)}$$

$$= \sum_{i\in\Omega_{u\otimes q}}\ \sum_{k\in\Omega_{u\otimes\overline{q}}}(-1)^{\#((i\oplus k)\otimes(q\oplus v))} \tag{3}$$

$$= \sum_{i\in\Omega_{u\otimes q}}(-1)^{\#(i\otimes q)}\sum_{k\in\Omega_{u\otimes\overline{q}}}(-1)^{\#(k\otimes v)} \tag{4}$$

$$= \sum_{k\in\Omega_{u\otimes\overline{q}}}(-1)^{\#(k\otimes v)} - \sum_{k\in\Omega_{u\otimes\overline{q}}}(-1)^{\#(k\otimes v)} = 0 \tag{5}$$

Equation (4) follows from (3) since $(i \oplus k) \otimes (q \oplus v) = (i\otimes q)\oplus(k\otimes v)$, and since $\#((i\otimes q)\oplus(k\otimes v)) \pmod 2 = \#(i\otimes q) + \#(k\otimes v) \pmod 2$. Equation (5) follows from (4) since $\Omega_q = \Omega_{u\otimes q} = \{0, q\}$. □

Proof of lemma 5: The simplex Λ is the convex hull of the basis vectors $e_0, e_1, \ldots, e_{N-1}$ of the standard basis. The vectors $\widehat{e}_0, \widehat{e}_1, \ldots, \widehat{e}_{N-1}$ are the same geometric points expressed in the Walsh basis, so the simplex is still the convex hull of these points. But these correspond to the columns of the Walsh matrix, and every entry of the Walsh matrix is $\pm 2^{-\ell/2}$. Thus $\widehat{x}_k$ is a convex combination of $2^{-\ell/2}$ and $-2^{-\ell/2}$. □

Proof of lemma 6:

$$\begin{aligned}\widehat{F}_{i,j} &= 2^{-\ell}\sum_{u\in\Omega}(-1)^{\#(u\otimes i)}\sum_{v\in\Omega}(-1)^{\#(v\otimes j)}F_{u,v}\\ &= 2^{-\ell}\sum_{u\in\Omega}(-1)^{\#(u\otimes i)}\sum_{v\in\Omega}(-1)^{\#(v\otimes j)}\delta_{u,v}F_{u,u}\\ &= 2^{-\ell}\sum_{u\in\Omega}(-1)^{\#(u\otimes i)}(-1)^{\#(u\otimes j)}f_u\\ &= 2^{-\ell}\sum_{u\in\Omega}(-1)^{\#(u\otimes(i\oplus j))}f_u\\ &= 2^{-\ell/2}\widehat{f}_{i\oplus j}\end{aligned}$$

□

Proof of lemma 12:

$$\begin{aligned}\widehat{f}_k &= 2^{-\ell/2}\sum_{i\in\Omega}(-1)^{\#(k\otimes i)}f_i\\ &= 2^{-\ell/2}c\sum_{i\in\Omega}(-1)^{\#(k\otimes i)}\\ &\qquad + 2^{-\ell/2}\sum_{j\in\mathcal{L}}b_j\sum_{i\in\Omega}(-1)^{\#(k\otimes i)}\delta_{i\otimes j,j}\\ &= 2^{\ell/2}c\delta_{k,0} + 2^{-\ell/2}\sum_{j\in\mathcal{L}}b_j\\ &\qquad \sum_{v\in\Omega_j}(-1)^{\#(k\otimes v)}\sum_{u\in\Omega_{\overline{j}}}(-1)^{\#(k\otimes u)}\delta_{(v\oplus u)\otimes j,j}\end{aligned}$$

Note that $(v\oplus u)\otimes j = v\otimes j$ and $\delta_{v\otimes j,j} = 1$ iff $v = j$. Thus,

$$\begin{aligned}\widehat{f}_k &= 2^{\ell/2}c\delta_{k,0} + 2^{-\ell/2}\sum_{j\in\mathcal{L}}b_j(-1)^{\#(k\otimes j)}\sum_{u\in\Omega_{\overline{j}}}(-1)^{\#(k\otimes u)}\\ &= 2^{\ell/2-1}\delta_{k,0}(2c+\sum_{j\in\mathcal{L}}b_j) - 2^{\ell/2-1}\sum_{j\in\mathcal{L}}\delta_{k,j}b_j\end{aligned}$$

The last statement of the lemma follows easily from the formula. □

Proof of Lemma 13: Let $\widehat{z}$ be a fixed point in the simplex.

In the special case where $\widehat{f}_k = 0$, $\widehat{z}_k = 0$, so throughout the rest of the proof we assume that $\widehat{f}_k < 0$.

Let $B = \frac{\mu}{1-2\mu}(\widehat{f}_0 + \sum_{j\in\mathcal{L}}\widehat{f}_j\widehat{z}_j)$. Solving equation (2) for $\widehat{z}_k$ gives:

$$\widehat{z}_k = \frac{1}{\widehat{f}_k}\left(-B \pm \sqrt{B^2+\widehat{f}_k^2}\right) \tag{6}$$

We claim that for a solution to be in the simplex, the plus sign must be used in equation (6). So assume that $\widehat{z}$ represents a solution in the simplex.

First we claim that $B > 0$. Recall that we assumed that $\widehat{f}_0 + \sum_{j\in\mathcal{L}}\widehat{f}_j \geq 0$ and $\widehat{f}_j < 0$ for all j. Since $\widehat{z}$ is in the simplex, and since mutation is positive, $\widehat{z}_j < 1$ for all $j\in\mathcal{L}$ by lemma 5. It follows easily that $B > 0$.

Now we assume that the minus sign is used in equation (6) and derive a contradiction. Thus,

$$\begin{aligned}\widehat{z}_k < 1 &\Longrightarrow \frac{1}{\widehat{f}_k}\left(-B-\sqrt{B^2+\widehat{f}_k^2}\right) < 1\\ &\Longrightarrow B+\sqrt{B^2+\widehat{f}_k^2} < -\widehat{f}_k\\ &\Longrightarrow \sqrt{B^2+\widehat{f}_k^2} < -\widehat{f}_k - B\\ &\Longrightarrow B^2+\widehat{f}_k^2 < \widehat{f}_k^2 + 2\widehat{f}_kB + B^2\\ &\Longrightarrow 0 < 2\widehat{f}_kB\end{aligned}$$

Since $\widehat{f}_k < 0$ and $B > 0$, this is a contradiction.

Thus, the plus sign must always be used in equation (6), and there is a unique fixed point. □

To prove theorem 14 we will use the Gershgorin Circle Theorem (page 685 of [2]).

Theorem 17 *Let J by an n by n real-valued matrix, and let λ be an eigenvalue of J. For some integer k,*

$$|J_{k,k}-\lambda| \leq \sum_{j\neq k}|J_{k,j}|$$

Proof of theorem 14: The existence and uniqueness of the fixed point was proved in lemma 13.

We calculate the differential of $\mathcal{G}$ from equation (1).

$$\frac{\partial\mathcal{G}(\widehat{z})_k}{\partial\widehat{z}_i} = (1-2\mu)\frac{\widehat{f}_k\widehat{f}_i\left(\widehat{z}_k^2-1\right)}{\left(\widehat{f}_0+\sum_j\widehat{f}_j\widehat{z}_j\right)^2}$$

$$\frac{\partial\mathcal{G}(\widehat{z})_k}{\partial\widehat{z}_k} = (1-2\mu)\frac{\left(\widehat{f}_0+\sum_{j\in\mathcal{L}\setminus\{k\}}\widehat{f}_j\widehat{z}_j\right)^2-\widehat{f}_k^2}{\left(\widehat{f}_0+\sum_j\widehat{f}_j\widehat{z}_j\right)^2} \tag{7}$$

Equation (2) can be rewritten in the form:

$$2\mu\widehat{z}_k\left(\widehat{f}_0+\sum_{j\in\mathcal{L}}\widehat{f}_j\widehat{z}_j\right) + (1-2\mu)\widehat{f}_k\left(\widehat{z}_k^2-1\right) = 0 \tag{8}$$

We claim that for a solution $\widehat{z}$ of these equations, $\widehat{z}_k \leq 0$ for all k. This follows from the equations since we have assumed that $\widehat{f}_k \leq 0$ and $\widehat{f}_0 + \sum_{j\in\mathcal{L}}\widehat{f}_j\widehat{z}_j \geq 0$, and we know that $|\widehat{z}_k| \leq 1$ since $\widehat{z}$ corresponds to a point in the simplex.

Assume that $\widehat{z}$ is a fixed point. Then using equation (8):

$$\frac{\partial \mathcal{G}(\widehat{z})_k}{\partial \widehat{z}_i} = \frac{-2\mu \widehat{f}_i \widehat{z}_k \left(\widehat{f}_0 + \sum_j \widehat{f}_j \widehat{z}_j\right)}{\left(\widehat{f}_0 + \sum_j \widehat{f}_j \widehat{z}_j\right)^2} = \frac{-2\mu \widehat{f}_i \widehat{z}_k}{\widehat{f}_0 + \sum_j \widehat{f}_j \widehat{z}_j}$$

We assumed earlier that $-\sum_j \widehat{f}_j \leq \widehat{f}_0$. Since $\widehat{z}$ corresponds to point in the simplex, $\widehat{z}_k \geq -1$, or $-\widehat{z}_i \leq 1$. Thus,

$$\begin{aligned} \sum_{i \in \mathcal{L} \setminus \{k\}} \left| \frac{\partial \mathcal{G}(\widehat{z})_k}{\partial \widehat{z}_i} \right| &= \frac{2\mu(-\widehat{z}_k)}{\widehat{f}_0 + \sum_j \widehat{f}_j \widehat{z}_j} \sum_{i \in \mathcal{L} \setminus \{k\}} (-\widehat{f}_i) \\ &\leq \frac{2\mu}{\widehat{f}_0 + \sum_j \widehat{f}_j \widehat{z}_j} \widehat{f}_0 < 2\mu \end{aligned}$$

Using equation (7),

$$\frac{\partial \mathcal{G}(\widehat{z})_k}{\partial \widehat{z}_k} \leq (1 - 2\mu)$$

The Gershgorin Circle Theorem shows that any eigenvalue of the differential is less that 1. □

Proof of Theorem 15: The relative error after t generations is:

$$\frac{|\mathcal{G}^t(0) - w_{fixed}|}{|w_{fixed}|} = \frac{\left| -\frac{C(A^t - 1)}{A-1} - \frac{C}{A-1} \right|}{\frac{C}{A-1}} = A^t$$

Further,

$$A^t < \epsilon \Longleftrightarrow t \ln A < \ln \epsilon \Longleftrightarrow t > \frac{-\ln \epsilon}{-\ln A}$$

Let $x = 1/\ell$ and let

$$\begin{aligned} f(x) = -\ln A &= -\ln \left(1 - \frac{2\alpha + 1}{\ell} + \frac{2\alpha}{\ell}\right) \\ &= -\ln(1 - (2\alpha + 1)x + 2\alpha x^2) \end{aligned}$$

The first-order Taylor series with remainder for f about $x = 0$ is

$$f(x) = (2\alpha + 1)x + \frac{-8\alpha^2\xi - 4\alpha\xi + 8\alpha^2\xi^2 + 4\alpha^2 + 1}{(1 - (2\alpha + 1)\,\xi + 2\alpha\xi^2)^2} \;\frac{x^2}{2}$$

where $0 < \xi < x$.

We claim that $B = -8\alpha^2\xi - 4\alpha\xi + 8\alpha^2\xi^2 + 4\alpha^2 + 1 \geq 0$. Make the substitution $\xi = \psi + \frac{2\alpha+1}{4\alpha}$. Then the expression becomes $8\alpha^2\psi^2 + \frac{1}{2}(4\alpha^2 - 4\alpha + 1)$ which is clearly positive.

Thus, $f(x) = (2\alpha + 1)x + B \geq (2\alpha + 1)x$, and

$$\frac{-\ell \ln \epsilon}{1 + 2\alpha} \geq \frac{-\ln \epsilon}{(1 - 2\alpha)\left(\frac{1}{\ell}\right) + B\left(\frac{1}{\ell}\right)^2} = \frac{-\ln \epsilon}{f\left(\frac{1}{\ell}\right)} = \frac{-\ln \epsilon}{-\ln A}$$

□

References

[1] H. Geiringer. On the probability of linkage in mendelian heredity. *Annals of Mathematical Statistics*, 15:25–57, 1944.

[2] Erwin Kreyszig. *Advanced Engineering Mathematics*. John Wiley and Sons, New York, third edition, 1972.

[3] Thilo Mahnig and Heinz Mühlenbein. Optimal mutation rate using bayesian priors for estimation of distribution algorithms. In K. Steinhüfel, editor, *Stochastic Algorithms: Foundations and Applications*, LNCS. Springer-Verlag, 2001.

[4] Heinz Mühlenbein. The equation for the response to selection and its use for prediction. *Evolutionary Computation*, 5(3):303–346, 1998.

[5] Heinz Mühlenbein and Thilo Mahnig. FDA – a scalable evolutionary algorithm for the optimization of additively decomposed functions. *Evolutionary Computation*, 7(4):353–376, 1999.

[6] Heinz Mühlenbein and Thilo Mahnig. Evolutionary algorithms: from recombination to search distributions. In L. Kallel, B. Naudts, and A. Rogers, editors, *Theoretical Aspects of Evolutionary Computation*, pages 137–176. Springer Verlag, 2000.

[7] C. R. Stephens and H. Waelbroeck. Effective degrees of freedom in genetic algorithms and the block hypothesis. In Thomas Back, editor, *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 34–40, San Mateo, 1997. Morgan Kaufman.

[8] H. Voigt and H. Mühlenbein. Gene pool recombination and the utilization of covariances for the breeder genetic algorithm. In Z. Michalewicz, editor, *Proc. of the 2nd IEEE International Conference on Evolutionary Computation*, pages 172– 177, New York, 1995. IEEE Press.

[9] M. D. Vose. *The Simple Genetic Algorithm: Foundations and Theory*. MIT Press, Cambridge, MA, 1999.

[10] M. D. Vose and A. H. Wright. Form invariance and implicit parallelism. *Evolutionary Computation*, 2001.

[11] A. H. Wright. The exact schema theorem. Technical report, University of Montana, Missoula, MT 59812, USA, 1999. http://www.cs.umt.edu/u/wright/.