The Simple Genetic Algorithm and the Walsh Transform: part I, Theory

Michael D. Vose
Computer Science Dept.
The University of Tennessee
Knoxville, TN 37996-1301
vose@cs.utk.edu

Alden H. Wright[1]
Computer Science Dept.
The University of Montana
Missoula, MT 59812-1008
wright@cs.umt.edu

ABSTRACT

This paper is the first part of a two part series. It proves a number of direct relationships between the Fourier transform and the simple genetic algorithm. (For a binary representation, the Walsh transform is the Fourier transform.) The results are of a theoretical nature and are based on the analysis of mutation and crossover. The Fourier transform of the mixing matrix is shown to be sparse. An explicit formula is given for the spectrum of the differential of the mixing transformation. By using the Fourier representation and the fast Fourier transform, one generation of the infinite population simple genetic algorithm can be computed in time $O(c^{\ell \log_2 3})$, where c is arity of the alphabet and ℓ is the string length. This is in contrast to the time of $O(c^{3\ell})$ for the algorithm as represented in the standard basis. There are two orthogonal decompositions of population space which are invariant under mixing. The sequel to this paper will apply the basic theoretical results obtained here to inverse problems and asymptotic behavior.

1 Introduction

That there is some sort of connection between Genetic Algorithms and the Walsh transform has been a part of GA folklore for a number of years. Folklore asserts that schema utilities are somehow put in a more perspicuous form via the Walsh transform, and, that schemata determine GA behavior via the schema theorem.

This perception may have been engendered by Bethke's Ph.D. dissertation (Bethke, 1981) where the Walsh transform was used as a tool to construct deceptive functions. Bethke's Walsh-schema analysis of functions is certainly beautiful on its own merits; schema fitness is expressed with remarkable compactness by Walsh coefficients and low order schemata require the least number of coefficients, facilitating analysis of deception by way of Walsh coefficients.

Deception has been and will continue to be an important concept in the analysis of genetic algorithm behavior. At the same time, deception is not necessarily germane in every circumstance

[1]Part of this work was done while the second author was visiting the Computer Science Department of the University of Tennessee.

(Grefenstette, 1993), deception is not necessarily correlated with high order Walsh coefficients (Goldberg, 1990), neither is the Walsh transform necessarily the appropriate tool for the construction or analysis of deceptive functions (Deb & Goldberg, 1993). Moreover, the degree to which schemata determine the course of genetic search via the schema theorem has been seriously called into question (Vose, 1993). The folklore connecting GAs and the Walsh transform is tenuous at best.

Traditionally, the Walsh transform has been applied to fitness. An early paper of Goldberg (1989) attempts to calculate the expected value of fitness, expressed in terms of Walsh coefficients, following the application of genetic operators. While to some extent successful, his paper is representative of early work in this area however; the effects of operators on Walsh coefficients may have been considered, but the analysis relies on heuristic arguments and does not involve the direct application of the Walsh transform to crossover and mutation, or to any of their associated mathematical objects. For example, Weinberger (1991) applies Fourier analysis to fitness landscapes and relates the Fourier coefficients to the autocorrelation function of a random walk on the landscape, but his work does not relate Fourier analysis to crossover operators. Aizawa (1997) extends earlier work on epistasis variance and crossover correlation by showing how the crossover correlation can be computed using Walsh coefficients, but she does not apply transform analysis to the mixing operators (mutation and recombination) themselves.

In contrast, our results are based on the direct application of the Walsh transform to mixing (mutation and recombination). At some stage the fitness function does get transformed, but that step is not the fulcrum of our analysis. It appears as an accommodation to the representation which arises naturally from other considerations related to mixing. Moreover, the connections we develop between the Walsh Transform and the genetic algorithm are compelling; they hold in every case (for any mutation, crossover, and fitness), they reveal fundamental structure (i.e., the eigenvalues of the mixing operator's differential), they provide the most efficient methods known to calculate with the infinite population model (in the general case), they provide the only method known to simulate evolution backwards in time, and they are proven as mathematical theorems.

Previous applications of the Walsh transform to mixing are, to our knowledge, sparse. The paper of Vose and Liepins (1991) was perhaps the first, demonstrating that the twist of the mixing matrix is triangularized by the Walsh transform. In a related paper, Koehler (1995) gives a congruence transformation defined by a lower triangular matrix that diagonalizes the mixing matrix for 1-point crossover and mutation given by a rate (the mixing matrix is defined at the end of section 2). The paper of J. N. Kok and P. Floreen (1995) is one of the more recent, independently obtaining several results which Vose presented at ICGA'95 in the advanced theory tutorial. Their paper also considers representing variance, representations by way of bit products, and nonuniform Walsh-schema transforms. Finally, Koehler, Bhattacharyya & Vose (1998) applies the Fourier transform to mixing in generalizing results concerning the simple genetic algorithm which were previously established for the binary case. That paper extends the analysis to strings over an alphabet of cardinality c, where c is an arbitrary integer greater than 1.

The goal of this paper and its sequel is to show how the Walsh transform appertains to the simple genetic algorithm in a natural and inherent way, particularly revealing the dynamics of mixing, and, as a special case, the dynamics of mutation. Through a series of results, a theoretical foundation will be laid which explains and exploits the interplay between the Walsh transform and the simple GA. This paper extends the previous account of our work (given in Vose & Wright, 1996) in three ways. First, it is far more complete, including details and explanations, second, it contains further results, and third, it is more general, providing a framework that extends directly to higher cardinality alphabets.

In a companion paper, "The Simple Genetic Algorithm and the Walsh Transform: part II, The Inverse", the theoretical groundwork developed here will be brought to bear on inverse problems and asymptotic behavior.

2 Basics

The formalism used is that of random heuristic search with heuristic $\mathcal{G}$ (see Vose & Wright, 1994, and Vose, 1996, the most comprehensive account is in Vose, 1998). This section reviews technical details, though the focus is on the case of fixed length c-ary strings. As first explained in Koehler, Bhattacharyya & Vose (1998), it is the *Fourier* transform, not the Walsh transform, that is appropriate in the general cardinality case (i.e., when $c > 2$). However, when $c = 2$ the Fourier transform *is* the Walsh transform; by working with the Fourier transform in the body of this paper we therefore implicitly deal with the Walsh transform while simultaneously providing a framework that extends directly to higher cardinality alphabets.

This paper explicitly deals with the Walsh transform by focusing on the binary case ($c = 2$) in the examples and the concrete results. The notation and several of the abstract results, however, will be stated in greater generality (for arbitrary c) to make plain how the analysis extends. Refined and completed by Vose, the extension of transform analysis to the general cardinality case began as joint work with Gary Koehler and Siddhartha Bhattacharyya (1998).

2.1 Notation

Square brackets $[\cdots]$ are, besides their standard use as specifying a closed interval of numbers, used to denote an indicator function: if *expr* is an expression which may be true or false, then

$$[\mathit{expr}] \quad = \quad \begin{cases} 1 & \text{if } \mathit{expr} \text{ is true} \\ 0 & \text{otherwise} \end{cases}$$

The search space Ω consists of c-ary strings of length ℓ. Let $n = c^\ell$. Integers in the interval $[0, n)$ are identified with elements of Ω through their c-ary representation. This correspondence allows them to be regarded as elements of the product group

$$\Omega \quad = \quad \underbrace{Z_c \times \cdots \times Z_c}_{\ell \text{ times}}$$

where Z_c denotes the integers modulo c. The group operation on this product (addition modulo c) is denoted by $\oplus$, and the operation of componentwise multiplication (modulo c) is denoted by $\otimes$. Componentwise subtraction (modulo c) is denoted by $\ominus$, and $0 \ominus x$ is abbreviated as $-x$. When elements of Ω are bit strings (i.e., $c = 2$), both $\oplus$ and $\ominus$ are the componentwise "exclusive-or" operation, and $\otimes$ is componentwise "and". The notation $\overline{k}$ abbreviates $1 \ominus k$. The operation $\otimes$ takes precedence over $\oplus$ and $\ominus$, and all three bind more tightly than other operations, except for $k \mapsto \overline{k}$ which is unary and has highest precedence.

An element k of Ω will also be thought of as a column vector in $\Re^\ell$ (its c-ary digits are the components), and in that case is represented with least significant c-ary digit at the top.

Angle brackets $<\cdots>$ denote a tuple, which is to be regarded as a column vector, $\mathrm{diag}(x)$ denotes the square diagonal matrix with iith entry x_i. Indexing always begins with zero. Superscript T denotes transpose, superscript C denotes complex conjugate, and superscript H denotes conjugate transpose. Let $\mathbf{1}$ denote the column vector of all 1's. The jth basis vector e_j is the jth column of the identity matrix. The space perpendicular to a vector v is $v^\perp$.

For example, if $c = 3$ and $\ell = 5$, then the string 21021 corresponds to the integer whose decimal representation is 196, and to the column vector $< 1 \;\; 2 \;\; 0 \;\; 1 \;\; 2 >$. In string notation, $-21021 = 12012$, $21021 \oplus 11220 = 02211$, $21021 \otimes 11220 = 21010$, and $21021 \ominus 11220 = 10101$.

Given $k \in \Omega$, let those i for which $k \otimes c^i > 0$ be $i_0 < i_1 < \cdots < i_{m-1}$ where $m = \#k$, and where $\#k$ denotes the number of nonzero c-ary digits of k. The *injection corresponding to k* is the $\ell \times m$ matrix K defined by $K_{i,j} = [i = i_j]$. To make explicit the dependence of Ω on the string length ℓ, it may be written as ${}^\ell\Omega$. The *embedding corresponding to k* is the image under K of ${}^m\Omega$ (regarding elements of ${}^m\Omega$ and ${}^\ell\Omega$ as column vectors) and is denoted Ω_k. Integers in the interval $[0, c^m)$ correspond to elements of Ω_k through K. Note that Ω_k is an Abelian (commutative) group (Birkhoff and MacLane, 1953) under the operation $\oplus$, and, more generally, is a commutative ring with respect to $\oplus$ and $\otimes$.

For example, consider the binary case ($c = 2$) with $\ell = 6$ and $k = 26 \equiv 011010$, which gives $m = 3$ and

$$K = \begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}$$

integer	element of ${}^3\Omega$	corresponding element of Ω_{26}
0	000	000000
1	001	000010
2	010	001000
3	011	001010
4	100	010000
5	101	010010
6	110	011000
7	111	011010

As illustrated above, embedding an element of ${}^m\Omega$ corresponds to distributing its bits among the locations where k is nonzero.

An element $k \in \Omega$ is called *binary* (even if $c > 2$) provided that $k_i > 0 \Longrightarrow k_i = 1$ (recall that k is naturally a column vector). The utility of embeddings follows from the fact that if k is binary then each $i \in \Omega$ has a unique representation $i = u \oplus v$ where $u \in \Omega_k$ and $v \in \Omega_{\overline{k}}$. This follows from the identity $i = i \otimes k \oplus i \otimes \overline{k} = u \oplus v$.

The symbol $\setminus$ denotes set subtraction ($A \setminus B$ is the set of elements in A which are not in B). The notation

$$f\Big|_B$$

represents the function f restricted to the domain B.

2.2 Selection

Define the *simplex* to be the set

$$\Lambda \;=\; \{<x_0, ..., x_{n-1}> : \; x_j \in \Re, \; x_j \geq 0, \; \Sigma x_j = 1\}$$

An element p of Λ corresponds to a population according to the rule

$$p_i \;=\; \text{the proportion of } i \text{ contained in the population}$$

The cardinality of each generation is a constant r called the *population size*. Hence the proportional representation given by p unambiguously determines a population once r is known. The vector p is referred to as a *population vector*.

Given a fitness function $f : \Omega \rightarrow \Re^+$, define the *fitness matrix* F to be the $n \times n$ diagonal matrix $F_{i,i} = f(i)$. Since f is positive, F is invertible. It follows that the function $\mathcal{F}(x) = Fx/\mathbf{1}^T Fx$ is also invertible and

$$\mathcal{F} \, : \, \Lambda \longrightarrow \Lambda$$

The image of a population vector p under $\mathcal{F}$ is called a *selection vector*. The ith component of $\mathcal{F}(p)$ is the probability with which i is to be selected (with replacement) from the current population p.

2.3 Mutation

The symbol μ will be used for three different (though related) things. This overloading of μ does not take long to get used to because context makes its meaning clear. The benefits are clean and elegant presentation and the ability to use a common symbol for ideas whose differences are often conveniently blurred.

First, $\mu \in \Lambda$ can be regarded as a *distribution* describing the probability μ_i with which i is selected to be a mutation mask (additional details follow).

Second, $\mu : \Omega \to \Omega$ can be regarded as a *mutation function* which is nondeterministic. The result $\mu(x)$ of applying μ to x is $x \oplus i$ with probability μ_i. The i occurring in $x \oplus i$ is referred to as a *mutation mask*.

Third, $\mu \in [0, 0.5)$ can be regarded as a *mutation rate* which implicitly specifies the distribution μ according to the rule

$$\mu_i \quad = \quad (\mu/(c-1))^{\#i}\,(1-\mu)^{\ell - \#i}$$

The distribution μ need not correspond to any mutation rate, although that is certainly the classical situation. Any element $\mu \in \Lambda$ whatsoever is allowed. The effect of mutation is to alter the positions of string x in those places where the mutation mask i is nonzero. For arbitrary $\mu \in \Lambda$, mutation is called *positive* if $\mu_i > 0$ for all i. Mutation is called *zero* if $\mu_i = [i = 0]$.

2.4 Crossover

It is convenient to use the concept of *partial probability* . Let $\zeta : A \to B$ and suppose $\phi : A \to [0, 1]$, where $\sum \phi(a) = 1$. To say "$\xi = \zeta(a)$ with partial probability $\phi(a)$" means that $\xi = b$ with probability $\sum_a [\zeta(a) = b]\, \phi(a)$.

The description of crossover parallels the description of mutation; the symbol χ will be used for three different (though related) things.

First, binary $\chi \in \Lambda$ can be regarded as a *distribution* describing the probability χ_i with which i is selected to be a crossover mask (additional details will follow).

Second, $\chi : \Omega \times \Omega \to \Omega$ can be regarded as a *crossover function* which is nondeterministic. The result $\chi(x, y)$ is $x \otimes i \oplus \bar{i} \otimes y$ with partial probability $\chi_i/2$ and is $y \otimes i \oplus \bar{i} \otimes x$ with partial probability $\chi_i/2$. The i occurring in the definition of $\chi(x, y)$ is referred to as a *crossover mask*. The application of $\chi(x, y)$ to x, y is referred to as *recombining* x and y.

The arguments x and y of the crossover function are called *parents*, the pair $x \otimes i \oplus \bar{i} \otimes y$ and $y \otimes i \oplus \bar{i} \otimes x$ are referred to as their *children*. Note that crossover produces children by exchanging the components of parents in those positions where the crossover mask i is 1. The result $\chi(x, y)$ is called their *child*. Thus, the probability that z results as a child from recombining parents x

and y is

$$\sum_i \left([z = x \otimes i \oplus \overline{i} \otimes y] \frac{\chi_i}{2} + [z = y \otimes i \oplus \overline{i} \otimes x] \frac{\chi_i}{2} \right) = \sum_i \frac{\chi_i + \chi_{\overline{i}}}{2} [z = x \otimes i \oplus \overline{i} \otimes y]$$

Third, $\chi \in [0,1]$ can be regarded as a *crossover rate* which specifies the distribution χ according to the rule

$$\chi_i \quad = \quad \begin{cases} \chi t_i & \text{if } \; i > 0 \\ 1 - \chi + \chi t_0 & \text{if } \; i = 0 \end{cases}$$

where binary $t \in \Lambda$ is referred to as the *crossover type*. Classical crossover types include *1-point crossover*, for which

$$t_i \quad = \quad \begin{cases} 1/(\ell - 1) & \text{if } \; \exists k \in (0, \ell), \;\; i = 2^k - 1 \\ 0 & \text{otherwise} \end{cases}$$

and *uniform crossover*, for which $t_i = 2^{-\ell}$. However, any binary $t \in \Lambda$ whatsoever is allowed as a crossover type.

2.5 The heuristic $\mathcal{G}$

The *simple genetic algorithm* is given by applying the heuristic corresponding to selection (twice) to produce x and y as parents, followed by mutation of x and y, followed by crossover of the results of mutation. The pair selected are called parents, and the end result (only one of the two strings resulting from crossover is kept) is their child. The *mixing matrix* M is defined by the probability that child 0 is obtained:

$$M_{x,y} \quad = \quad \sum_{i,j,k} \mu_i \, \mu_j \, \frac{\chi_k + \chi_{\overline{k}}}{2} [(x \oplus i) \otimes k \oplus \overline{k} \otimes (y \oplus j) = 0]$$

The probability that child u is obtained from parents x and y is $M_{x \ominus u, y \ominus u}$, which follows from the following

$$\begin{aligned} & [((x \ominus u) \oplus i) \otimes k \oplus \overline{k} \otimes ((y \ominus u) \oplus j) = 0] \\ = \;\; & [((x \oplus i) \otimes k \oplus \overline{k} \otimes (y \oplus j)) \ominus (u \otimes k \oplus \overline{k} \otimes u) = 0] \\ = \;\; & [(x \oplus i) \otimes k \oplus \overline{k} \otimes (y \oplus j) \ominus u = 0] \\ = \;\; & [(x \oplus i) \otimes k \oplus \overline{k} \otimes (y \oplus j) = u] \end{aligned}$$

Under very general conditions, it does not matter whether mutation is preformed before or after crossover, because the mixing matrix would be the same (see Koehler, Bhattacharyya & Vose, 1998).

For example, a 2-bit representation ($c = 2$ and $\ell = 2$) with uniform crossover with rate χ and mutation rate μ has mixing matrix M given by the following symmetric matrix (only the upper half is shown)

$$\begin{pmatrix} (1-\mu)^2 & (1-\mu)/2 & (1-\mu)/2 & \frac{1}{2}-(\mu-\mu^2)(1-\chi)-\chi/4 \\ * & \mu-\mu^2 & (\mu-\mu^2)(1-\chi)+\chi/4 & \mu/2 \\ * & * & \mu-\mu^2 & \mu/2 \\ * & * & * & \mu^2 \end{pmatrix}$$

Let σ_k be the permutation matrix with i,j th entry given by $[j \ominus i = k]$. Then $(\sigma_k x)_i = x_{i \oplus k}$. Define the *mixing scheme* $\mathcal{M} : \Lambda \to \Lambda$ by

$$\mathcal{M}(x) = <\ldots, (\sigma_i x)^T M \sigma_i x, \ldots>$$

The i th component function $\mathcal{G}_i$ of the simple genetic algorithm's heuristic is the probability that i is the end result of selection, mutation, and crossover. In vector form it is

$$\mathcal{G}(p) = \mathcal{M} \circ \mathcal{F}(p)$$

where p is the current population vector.

3 The Fourier Transform

To streamline notation, let $e(x)$ abbreviate $e^{2\pi\sqrt{-1}\,x/c}$. The *Fourier matrix* is defined by

$$W_{i,j} = n^{-1/2} e(i^T j)$$

The Fourier matrix is symmetric and unitary ($W^H = W^C = W^{-1}$, where superscript C denotes complex conjugate, and superscript H denotes conjugate transpose). The *Fourier transform* is the mapping $x \mapsto Wx^C$. When $c = 2$, all objects are real, the conjugation may therefore be dispensed with, and the Fourier transform reduces to the Walsh transform determined by the matrix

$$W_{i,j} = n^{-1/2} (-1)^{i^T j}$$

In order to keep formulas simple, it is helpful, for matrix A and vector x, to represent $WA^C W^C$ and Wx^C concisely. The former is denoted by $\widehat{A}$ and the latter by $\widehat{x}$. The matrix $\widehat{A}$ is referred to as the Fourier transform of the matrix A. If y is a *row vector*, then $\widehat{y}$ denotes $y^C W^C$ (which is referred to as the Fourier transform of y).

3.1 Basic Properties

Let x be a column vector, y a row vector, and A a square matrix. If u and v are any of these, then $\widehat{\widehat{u}} = u$, $\widehat{u+v} = \widehat{u} + \widehat{v}$, $\widehat{uv} = \widehat{u}\widehat{v}$, $\widehat{u^H} = \widehat{u}^H$, $\widehat{u^{-1}} = \widehat{u}^{-1}$, $\text{spec}(\widehat{u}) = \text{spec}(u^C)$ whenever operations are defined. These properties follow easily from the definitions. Moreover if A is symmetric, then $\widehat{A}_{i,j} = \widehat{A}_{-j,-i}$.

Define the *twist* A^* of a $n \times n$ matrix A by

$$(A^*)_{i,j} \quad = \quad A_{j \ominus i, -i}$$

Let id denote the identity operator $A \mapsto A$, let $*$ denote the twist operator $A \mapsto A^*$, let H denote the conjugate transpose operator $A \mapsto A^H$, and let $\wedge$ denote the transform operator $A \mapsto \widehat{A}$. Vose (1998) has proved that the set of operators $\{H, \wedge, *\}$ generate a group under composition (the algebraic convention of applying operators from left to right is followed) with relations $\wedge H = H \wedge$, $H * H = ** = \wedge * \wedge$, and $HH = \wedge\wedge = *** = id$. Though tedious, this may be verified in a straightforward manner by direct calculation.

The following theorem is one of the key links between the simple genetic algorithm and the Fourier transform. Another, as will be seen later, is the effect that the Fourier transform has on the mixing matrix M and on its twist M^*.

Theorem 3.1

$$\widehat{\sigma_k} \quad = \quad \sqrt{n}\ diag(\widehat{e_{-k}})$$

Sketch of proof:

$$\begin{aligned}
(\widehat{\sigma_k})_{i,j} \quad &= \quad n^{-1} \sum_{u,v} e(i^T u)\, [v \ominus u = k]\, e(-v^T j) \\
&= \quad n^{-1} \sum_{u} e(i^T u)\, e(-(k+u)^T j) \\
&= \quad n^{-1/2}\, e(-k^T j)\, n^{-1/2} \sum_{u} e(u^T (i-j)) \\
&= \quad W_{-k,j}\, \sqrt{n} \sum_{u} W_{-u,0}\, W_{u,i \ominus j} \\
&= \quad \sqrt{n}\, W_{-k,j}\, (W^H W)_{0,i \ominus j} \\
&= \quad \sqrt{n}\, W_{-k,j}\, [i = j]
\end{aligned}$$

□

The group $\{\sigma_k\}$ of permutation matrices is *inherently* related to mixing through the definition

$$\mathcal{M}(x)_k = (\sigma_k x)^T M (\sigma_k x)$$

As described in (Vose, 1990), it follows that $\{\sigma_k\}$ is the linear group which commutes with mixing,

$$\sigma_k \mathcal{M}(x) = \mathcal{M}(\sigma_k x)$$

A crucial property of the Fourier transform is that it simultaneously diagonalizes this linear group (this is theorem 3.1). In fact, this property can be used to obtain the Walsh transform as follows. A theorem of linear algebra states that if a family of normal matrices commutes, then they are simultaneously diagonalizable by a similarity transformation (Gantmacher, 1977). Using a constructive proof of this theorem, a unitary matrix may be computed for the similarity transformation. Vose (1998) has done this in the binary case for the family $\{\sigma_k\}$, obtaining the Walsh matrix as the result. In other words, the Walsh transform can be obtained via fundamental symmetries which are embedded in the very definition of mixing. This generalizes to the general cardinality case; the columns of the Fourier matrix are related to fundamental symmetries which are embedded in the very definition of mixing, they form an orthonormal system of eigenvectors for the family $\{\sigma_k\}$ for general $c > 0$.

As might be suspected, this inherent relationship of the Fourier transform to mixing (theorem 3.1) indicates a fruitful direction to explore. The following section demonstrates the amazing ability of the Fourier transform to unravel the complexity of mixing. A consequence (to be considered later) is an explicit formula for the spectrum of the differential of the mixing scheme.

3.2 Applications to M

Theorem 3.2 $$\widehat{M}_{-x,y} \;=\; [x^T y = 0]\,\frac{n}{2}\,\widehat{\mu}_x\,\widehat{\mu}_y \sum_k (\chi_k + \chi_{\overline{k}})\,[x \otimes \overline{k} = 0 \,\wedge\, y \otimes k = 0]$$

Sketch of proof:

$$\begin{aligned}(WM^C W^C)_{-x,y} &= \frac{1}{2n} \sum_{u,v} e(-x^T u - v^T y) \sum_{i,j,k} \mu_i\,\mu_j\,(\chi_k + \chi_{\overline{k}})\,[(u \oplus i) \otimes k \oplus \overline{k} \otimes (v \oplus j) = 0] \\ &= \frac{1}{2n} \sum_{i,j,k} \mu_i\,\mu_j\,(\chi_k + \chi_{\overline{k}}) \sum_{u,v} e(-x^T u - y^T v)\,[(u \oplus i) \otimes k = -\overline{k} \otimes (v \oplus j)]\end{aligned}$$

The condition of the characteristic function is equivalent to $k \otimes (u \oplus i) = \overline{k} \otimes (v \oplus j) = 0$. Hence the innermost sum is

$$\begin{aligned}\sum_{k \otimes u = 0} e(-x^T u \ominus i) \sum_{\overline{k} \otimes v = 0} e(-y^T v \ominus j) &= e(x^T i + y^T j) \sum_{u \,\in\, \Omega_{\overline{k}}} e(-x^T u) \sum_{v \,\in\, \Omega_k} e(-y^T v) \\ &= n\, e(x^T i + y^T j)\,[x \otimes \overline{k} = 0 \,\wedge\, y \otimes k = 0]\end{aligned}$$

Incorporating these simplifications yields

$$\begin{aligned}\widehat{M}_{-x,y} &= \frac{1}{2} \sum_{i,j} \mu_i\,\mu_j\,e(x^T i + y^T j) \sum_k (\chi_k + \chi_{\overline{k}})\,[x \otimes \overline{k} = 0 \,\wedge\, y \otimes k = 0] \\ &= \frac{n}{2}\,\widehat{\mu}_x\,\widehat{\mu}_y \sum_k (\chi_k + \chi_{\overline{k}})\,[x \otimes \overline{k} = 0 \,\wedge\, y \otimes k = 0]\end{aligned}$$

$\square$

Note what has been accomplished by theorem 3.2. The mixing matrix M, which is dense when mutation is positive, has a sparse Fourier transform! The only entries $\widehat{M}_{x,y}$ which can be nonzero are those for which $[x^T y = 0]$. Moreover, the number of these is

$$\begin{aligned}
\sum_x c^{\ell-\#x} &= \sum_k c^{\ell-k} \sum_x [\#x = k] \\
&= \sum_k c^{\ell-k} \begin{pmatrix} \ell \\ k \end{pmatrix} (c-1)^k \\
&= c^\ell \sum_k \begin{pmatrix} \ell \\ k \end{pmatrix} (1 - \frac{1}{c})^k \\
&= c^\ell (2 - 1/c)^\ell
\end{aligned}$$

Therefore, the proportion of nonzero entries is $(2/c - 1/c^2)^\ell$, which converges to zero exponentially fast as ℓ increases (since $c \geq 2$).

The following two corollaries are fairly straightforward consequences (for detailed proofs, see Koehler, Bhattacharyya & Vose, 1998).

Corollary 3.3 *If mutation is zero, then* $M = \widehat{M}$.

Corollary 3.4 $\widehat{M^*}$ *is lower triangular. If mutation is zero, then* M^* *is upper triangular.*

Theorem 3.2 has further theoretical implications which will be explored later on. Note what has been accomplished by corollary 3.4. Direct access to the spectrum of M^* has been obtained since M^* is triangularized by taking its transform (and $\text{spec}(\widehat{M^*}) = \text{spec}(M^*)^C$). We will also see later how M^* is a crucial component of the differential $d\mathcal{M}$ of the mixing scheme $\mathcal{M}$.

The present goal, however, is to demonstrate the utility of the Walsh transform in simplifying the representation of a concrete mixing matrix. Towards that end, the following lemma will be useful. For the remainder of this subsection (section 3.2) the binary case is assumed (i.e. $c = 2$).

Lemma 3.5 $\displaystyle\sum_j (-1)^{x^T j} \alpha^{\mathbf{1}^T j} \quad = \quad (1-\alpha)^{\mathbf{1}^T x} (1+\alpha)^{\mathbf{1}^T \overline{x}}$

Sketch of proof: Inducting on $\mathbf{1}^T x$, the base case follows from the binomial theorem. Let $\overline{k} = 2^i$ be such that $\overline{k} \otimes x > 0$. Write x as $x = k \otimes x \oplus \overline{k} = K K^T x \oplus \overline{k} = K y \oplus \overline{k}$, and write j as $j = u \oplus v$ where $u \in \Omega_{\overline{k}} = \{0, \overline{k}\}$ and $v \in \Omega_k$. Note that $y \in {}^{\ell-1}\Omega$. The left hand side of the proposition is

$$\sum_{j \in {}^\ell\Omega} (-1)^{x^T j} \alpha^{\mathbf{1}^T j} \quad = \quad \sum_{u \in \{0, \overline{k}\}} (-1)^{x^T u} \alpha^{\mathbf{1}^T u} \sum_{v \in \Omega_k} (-1)^{x^T v} \alpha^{\mathbf{1}^T v}$$

$$= \quad (1-\alpha) \sum_{v \,\in\, K({}^{\ell-1}\Omega)} (-1)^{y^T K^T v}\, \alpha^{v^T \mathbf{1}}$$

$$= \quad (1-\alpha) \sum_{w \,\in\, {}^{\ell-1}\Omega} (-1)^{y^T K^T K w}\, \alpha^{w^T K^T \mathbf{1}}$$

$$= \quad (1-\alpha) \sum_{w \,\in\, {}^{\ell-1}\Omega} (-1)^{y^T w}\, \alpha^{w^T \mathbf{1}}$$

Applying the inductive hypothesis (since $\mathbf{1}^T y < \mathbf{1}^T x$) completes the proof. □

Proposition 3.6 *If mutation is affected by a mutation rate, then* $\widehat{\mu}_x \;=\; 2^{-\ell/2}\,(1-2\mu)^{\mathbf{1}^T x}$

Sketch of proof:

$$2^{\ell/2}\,\widehat{\mu}_x \quad = \quad \sum_j (-1)^{x^T j}\,\mu^{\mathbf{1}^T j}\,(1-\mu)^{\ell-\mathbf{1}^T j} \quad = \quad (1-\mu)^\ell \sum_j (-1)^{x^T j} \left(\frac{\mu}{1-\mu}\right)^{\mathbf{1}^T j}$$

Applying proposition 3.5 to the right hand side completes the proof. □

The next proposition, which handles the remaining factor in theorem 3.2 for the case of 1-point crossover, relies on the following auxiliary functions

$$\mathrm{hi}(x) \quad = \quad \begin{cases} 0 & \text{if } x = 0 \\ \sup\{i : 2^i \otimes x > 0\} & \text{otherwise} \end{cases}$$

$$\mathrm{lo}(x) \quad = \quad \begin{cases} \ell & \text{if } x = 0 \\ \inf\{i : 2^i \otimes x > 0\} & \text{otherwise} \end{cases}$$

Intuitively, the function $\mathrm{hi}(x)$ returns the index of the high order bit of x, and $\mathrm{lo}(x)$ returns the index of the low order bit. The notation $(\mathit{expr})^+$, used in the next proposition, denotes $\max\{0,\ \mathit{expr}\}$.

Proposition 3.7 *For 1-point crossover, if* $x \otimes y = 0$ *, then*

$$\sum_{k \,\in\, \Omega_{\overline{x} \otimes \overline{y}}} \chi_{k \oplus x} \quad = \quad (1-\chi)\,\delta_{x,0} \;+\; \frac{\chi}{\ell-1}\,(\mathrm{lo}(y) - \mathrm{hi}(x))^+$$

Sketch of proof: Let m denote a variable over the domain $\{2^i - 1 : 0 < i < \ell\}$. Note that for a given value of m there is a corresponding value of i, and vice versa. It follows from the definition of the 1-point crossover type that the left hand side of proposition 3.7 is

$$\sum_{k \in \Omega_{\overline{x} \otimes \overline{y}}} (1-\chi)\, \delta_{x \oplus k, 0} \; + \; \frac{\chi}{\ell - 1} \, [\exists m \, . \, x \oplus k = m]$$

$$= \quad (1-\chi)\, \delta_{x,0} \; + \; \frac{\chi}{\ell - 1} \sum_{m,k} [x \oplus k = m \, \wedge \, k \otimes (\mathbf{1} \oplus \overline{x} \otimes \overline{y}) = 0]$$

Note that

$$x \otimes y = 0 \, \wedge \, x \oplus k = m \implies k \otimes (\mathbf{1} \oplus \overline{x} \otimes \overline{y}) \; = \; (x \oplus m) \otimes (x \oplus y) \; = \; x \otimes \overline{m} \oplus m \otimes y$$

Hence the characteristic function in the sum above simplifies to $[x \in \Omega_m \; \wedge \; y \in \Omega_{\overline{m}}]$. The observation

$$\sum_m [x \in \Omega_m \; \wedge \; y \in \Omega_{\overline{m}}] \quad = \quad \sum_{\mathrm{hi}(x) < i} [y \in \Omega_{\overline{m}}] \quad = \quad (\mathrm{lo}(y) - \mathrm{hi}(x))^+$$

completes the proof. □

Collecting together the previous results yields

Proposition 3.8 *For 1-point crossover and mutation rate μ, $\widehat{M}_{i,j}$ is given by*

$$\delta_{i \otimes j, 0}\, (1-2\mu)^{\mathbf{1}^T i \oplus j} \Big((1-\chi)\, \frac{\delta_{i,0} + \delta_{j,0}}{2} \; + \; \frac{\chi}{\ell - 1} \, \frac{(\mathrm{lo}(j) - \mathrm{hi}(i))^+ \; + \; (\mathrm{lo}(i) - \mathrm{hi}(j))^+}{2} \Big)$$

The special case of proposition 3.8 corresponding to $i = 0$ was first proved by Gary J. Koehler (1994) (this has since been generalized, see Koehler, Bhattacharyya & Vose, 1998). Proposition 3.6 is also due to him (though by a much more complicated argument). Proposition 3.8 is noted here as a concrete example of how the Walsh transform drastically simplifies representation.

For example, the Walsh transform of the (2-bit representation) mixing matrix given earlier is

$$\widehat{M} = \begin{pmatrix} 1 & \frac{1}{2} - \mu & \frac{1}{2} - \mu & (\frac{1}{2} - \mu)^2 (2 - \chi) \\ \frac{1}{2} - \mu & 0 & (\frac{1}{2} - \mu)^2 \chi & 0 \\ \frac{1}{2} - \mu & (\frac{1}{2} - \mu)^2 \chi & 0 & 0 \\ (\frac{1}{2} - \mu)^2 (2 - \chi) & 0 & 0 & 0 \end{pmatrix}$$

As predicted (see the discussion preceding corollary 3.3), $\widehat{M}$ is sparse, containing 9 nonzero entries. The Walsh transform of the twist of the (2-bit representation) mixing matrix given earlier is given by

$$\widehat{M^*} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2}-\mu & \frac{1}{2}-\mu & 0 & 0 \\ \frac{1}{2}-\mu & 0 & \frac{1}{2}-\mu & 0 \\ \left(\frac{1}{2}-\mu\right)^2(2-\chi) & \left(\frac{1}{2}-\mu\right)^2\chi & \left(\frac{1}{2}-\mu\right)^2\chi & \left(\frac{1}{2}-\mu\right)^2(2-\chi) \end{pmatrix}$$

As predicted by corollary 3.4, it is lower triangular.

3.3 Applications to $d\mathcal{M}$

The main application of theorem 3.2 is to the differential $d\mathcal{M}$ of the mixing scheme $\mathcal{M}$. Some preliminary observations prepare the way.

Observe that $(\sigma_u^T M^* \sigma_u)_{i,j} = (M^*)_{i\ominus u,j\ominus u} = M_{j\ominus i,u\ominus i}$. Next note that the i,j th entry of $d\mathcal{M}_x$ is given by

$$\frac{\partial}{\partial x_j} \sum_{u,v} x_u\, x_v\, M_{u\ominus i,v\ominus i} \quad = \quad \sum_{u,v} (\delta_{u,j}x_v + \delta_{v,j}x_u)\, M_{u\ominus i,v\ominus i} \quad = \quad 2\sum_u x_u\, M_{j\ominus i,u\ominus i}$$

This establishes

Theorem 3.9 $d\mathcal{M}_x = 2\sum \sigma_u^T\, M^*\, \sigma_u\, x_u$

Notice how the twist M^* of the mixing matrix, as well as the permutation group $\{\sigma_u\}$, appear naturally in the differential as given by theorem 3.9. In view of theorem 3.1 and corollary 3.4, one would expect the Fourier transform to be particularly revealing of properties of the mixing scheme $\mathcal{M}$. We will see that is the case to some extent below, and the sequel to this paper will demonstrate it in full.

The next proposition is a stepping stone towards determining the spectrum of $d\mathcal{M}_x$, and will also be useful when the Fourier basis is considered in the next section.

Proposition 3.10 $(\widehat{d\mathcal{M}_x})_{i,j} \quad = \quad 2\sqrt{n}\,(\widehat{M^*})_{i,j}\; \widehat{x}_{i\ominus j}$

Sketch of proof: By theorem 3.1, $(\widehat{\sigma_u})_{i,j} = \delta_{i,j}\, e(-j^T u)$, so by theorem 3.9 and the properties of the transform listed in section 3.1,

$$\begin{aligned} (\widehat{d\mathcal{M}_x})_{i,j} \quad &= \quad 2\sum_u (\widehat{\sigma_u}^C\, \widehat{M^*}\, \widehat{\sigma_u})_{i,j}\, x_u \\ &= \quad 2\sum_u x_u \sum_{v,w} \delta_{i,v}\, e(u^T i)\, (\widehat{M^*})_{v,w}\, \delta_{w,j} e(-j^T u) \end{aligned}$$

$$= \quad 2\sum_u x_u \, e(u^T(i-j)) \, (\widehat{M^*})_{i,j}$$

$$= \quad 2\sqrt{n} \, (\widehat{M^*})_{i,j} \, (Wx)_{i \ominus j}$$

$\square$

The next proposition is useful because the mixing scheme $\mathcal{M}$ is defined with respect to an ambient space ($\Re^n$) of dimension larger than its domain Λ. The differential of $\mathcal{M}$ at x is the unique linear transformation $d\mathcal{M}_x$ satisfying

$$\mathcal{M}(x+y) \quad = \quad \mathcal{M}(x) \; + \; d\mathcal{M}_x y \; + \; o(y)$$

This relation indicates that $x+y$ and x are in the domain of $\mathcal{M}$, hence $\mathbf{1}^T(x+y) = \mathbf{1}^T x = 1$. It follows that $y \in \mathbf{1}^\perp$. Therefore, $d\mathcal{M}_x\,|_{\mathbf{1}^\perp}$ is the relevant linear map (unless a domain larger than Λ is being considered). The following result helps clarify what happens when $d\mathcal{M}$ is restricted to $\mathbf{1}^\perp$.

Proposition 3.11 *Suppose $Ax = \lambda x$. Then*

$$\operatorname{spec}(A) \quad = \quad \operatorname{spec}(A^T \Big|_{x^\perp}) \; \cup \; \{\lambda\}$$

Sketch of proof: Since x is an eigenvector of A, it follows that $A^T : x^\perp \to x^\perp$. Without loss of generality x is a unit vector. Let $\{b_1, \ldots, b_n\}$ be an orthonormal basis with $b_n = x$, and let these vectors form the columns of B. Observe that if $j < n$ then $B^T A^T B \, e_j \; = \; B^{-1} A^T b_j \; \subset \; B^{-1}(x^\perp) \; \subset \; e_n^\perp$. Hence

$$B^T A^T B \quad = \quad \begin{pmatrix} C & * \\ 0 & * \end{pmatrix} \qquad \text{and} \qquad B^T A B \quad = \quad \begin{pmatrix} C^T & 0 \\ * & * \end{pmatrix}$$

Note that $B^T A B \, e_n = B^T A x = \lambda B^{-1} b_n = \lambda e_n$. Thus the last diagonal element in the matrices above is λ. Since, with respect to the chosen basis, elements of $x^\perp$ have the form

$$\begin{pmatrix} * \\ 0 \end{pmatrix}$$

it follows that C represents A^T on $x^\perp$. Since the spectrum is invariant under change of basis, it follows from the representation for $B^T A^T B$ that $\operatorname{spec}(A^T) = \operatorname{spec}(C) \cup \{\lambda\}$. The observation that a matrix has the same spectrum as its transpose completes the proof. $\square$

Because it is of potential interest to consider $\mathcal{M}$ on a domain larger than Λ, the following result is stated in greater generality.

Theorem 3.12 *The spectrum of $d\mathcal{M}_x$ is the spectrum of M^* multiplied by $2 \cdot \mathbf{1}^T x$. In particular, for $x \in \Lambda$, it is independent of x and is given by 2 times the 0 th column of $\widehat{M}$. If moreover mutation is positive, then the largest eigenvalue is 2 and all other eigenvalues are in the interior of the unit disk.*

Sketch of proof: Because the spectrum is invariant under conjugation by W (a change of basis), it suffices to consider the spectrum of $\widehat{d\mathcal{M}_x}$. By corollary 3.4 and proposition 3.10, $\widehat{d\mathcal{M}_x}$ is lower triangular, having spectrum given by its diagonal entries which are $2\sqrt{n}\,(\widehat{M^*})_{i,i}\,\widehat{x}_0$. By a straightforward computation $2\,(\widehat{M^*})_{i,i}\,\sqrt{n}\,\widehat{x}_0 \;=\; 2\,\widehat{M}_{-i,0}\;\mathbf{1}^T x$ (use $*\wedge = \wedge **$), which establishes the first part of theorem 3.12.

Since $(M^*)^T\mathbf{1} = \mathbf{1}$, it follows from proposition 3.11 that

$$\operatorname{spec}(M^*\Big|_{\mathbf{1}^\perp}) \quad = \quad \operatorname{spec}(M^*) \;\setminus\; \{1\}$$

As noted above, $(\widehat{M^*})_{i,i} = \widehat{M}_{-i,0}$, so, by theorem 3.2, the spectral radius of M^* restricted to $\mathbf{1}^\perp$ is

$$\sup_{j>0} \;\; \sqrt{n}\,\widehat{\mu}_j\,\frac{1}{2}\sum_k (\chi_k + \chi_{\overline{k}})\,[k \otimes j = 0]$$

If μ is positive and $j > 0$, cancellation occurs in the sum defining $\sqrt{n}\,\widehat{\mu}_j$ and so it must have modulus less than 1. Next note that the subscripts in the sum above are of the form u and $v \oplus j$ where $u, v \in \Omega_{\overline{j}}$. Since $\Omega_{\overline{j}}$ is a group, $u = v \oplus j$ is impossible; it would lead to the contradiction $u \ominus v = j \in \Omega_{\overline{j}}$. The sum can therefore have no repeated terms and is at most 1. Hence the spectral radius of M^* restricted to $\mathbf{1}^\perp$ is less than 1/2. $\square$

To summarize some of the most notable results of this section,

- Theorem 3.2 shows how the Fourier transform simplifies M in the general case.
- Proposition 3.8 specializes this to the Walsh transform in the concrete binary case corresponding to one-point crossover and mutation given by a rate.
- Corollary 3.4 shows the Fourier transform triangulates M^*.
- Theorem 3.12 reveals the spectrum of $d\mathcal{M}$ as a column of $2\widehat{M}$.

As has been explained in this section, the Fourier transform emerges in a natural way when one considers mixing. There are further connections however, as the following sections will show.

4 The Fourier Basis

In the binary case ($c = 2$), the hyperplane containing Λ is a translate in the direction of the first column of W of the linear span of the other columns. This observation hints that a natural basis for representing $\mathcal{G}$ might be given by the columns of W (i.e., $\widehat{e_0}, ..., \widehat{e_{n-1}}$). Moreover, and far more telling, the results of the previous section demonstrate how the Fourier transform – which essentially corresponds to a change of basis – profoundly simplifies M in the general case. The development of how $\mathcal{G}$ transforms in this representation (i.e., with respect to this basis) is the subject of this section.

The approach taken is through the differential of $\mathcal{M}$. The following theorem explains how $d\mathcal{M}$ transforms the Fourier basis $\mathcal{B} = \{\widehat{e_0}, ..., \widehat{e_{n-1}}\}$.

Theorem 4.1 $d\mathcal{M}_{\widehat{e_j}}\widehat{e_i} \;=\; 2\sqrt{n}\,\widehat{M}_{i,-j}\;\widehat{e_{i\oplus j}}$ *and* $\widehat{e_i}^T d\mathcal{M}_{\widehat{e_j}} \;=\; 2\sqrt{n}\,\widehat{M^*}_{i,-j}\;\widehat{e_{i\oplus j}}^T$

Sketch of proof: By proposition 3.10 and the properties of the transform listed in section 3.1,

$$\begin{aligned}
d\mathcal{M}_{\widehat{e_j}}\widehat{e_i} &= W(W^C d\mathcal{M}_{\widehat{e_j}} W\, e_i)^{CC} \\
&= W\widehat{d\mathcal{M}_{\widehat{e_j}}}^{C} e_i \\
&= 2\sqrt{n}\, W \sum_k (\widehat{M^*})^C_{k,i}\, \delta_{k\ominus i,j}\, e_k \\
&= 2\sqrt{n}\, (\widehat{M^*})^C_{i\oplus j,i}\;\; W e_{i\oplus j}
\end{aligned}$$

The proof of the first equation is completed by observing, when applied to a real symmetric matrix, that $* \wedge C = * \wedge HT = H ** \wedge T = ** \wedge T = \wedge * T$. Therefore $(\widehat{M^*})^C_{i\oplus j,i} = \widehat{M}_{j,-i} = \widehat{M}_{i,-j}$. The second equation is a consequence of the first,

$$\begin{aligned}
\widehat{e_i}^T d\mathcal{M}_{\widehat{e_j}} &= \widehat{e_i}^T d\mathcal{M}_{\widehat{e_j}} W\, (\sum_k e_k\, e_k^T)\, W^C \\
&= \widehat{e_i}^T d\mathcal{M}_{\widehat{e_j}} \sum_k \widehat{e_k}\, \widehat{e_{-k}}^T \\
&= \sum_k \widehat{e_i}^T\, (d\mathcal{M}_{\widehat{e_j}}\, \widehat{e_k})\, \widehat{e_{-k}}^T \\
&= 2\sqrt{n} \sum_k \widehat{M}_{k,-j}\, \widehat{e_i}^T \widehat{e_{k\oplus j}}\, \widehat{e_{-k}}^T \\
&= 2\sqrt{n}\, \widehat{M}_{j,i\oplus j}\, \widehat{e_{i\oplus j}}^T
\end{aligned}$$

The proof is completed by using the relation $\wedge = * \wedge *$. $\square$

Theorem 4.2 *$d\mathcal{M}_x\, y$ is symmetric and linear in x and y. Moreover,*

$$\begin{aligned}
\mathcal{M}(x) &= \frac{1}{2}\, d\mathcal{M}_x\, x \\
\mathcal{M}(x) \;-\; \mathcal{M}(y) &= d\mathcal{M}_{\frac{x+y}{2}}\, (x-y) \\
\mathbf{1}^T d\mathcal{M}_x &= 2\, \mathbf{1}^T x\, \mathbf{1}^T \\
\mathbf{1}^T \mathcal{M}(x) &= (\mathbf{1}^T x)^2
\end{aligned}$$

Sketch of proof: Symmetry in x and y follows from linearity and the fact that symmetry holds on a basis (via theorem 4.1; keep in mind that since M is symmetric, $\widehat{M}_{i,-j} = \widehat{M}_{j,-i}$). Linearity is a consequence of theorem 3.9. The second formula is a consequence of symmetry, linearity, and the first. The third formula follows from theorem 3.9 and a simple calculation (use the fact that $\mathbf{1}^T M^* = \mathbf{1}^T$). The last formula follows from the third and first. The first formula is a consequence of theorem 3.9 and a simple calculation:

$$\begin{aligned}
\mathcal{M}(x) &= \sum_i e_i \sum_k x_k \sum_j M_{j\ominus i, k\ominus i}\, x_j \\
&= \sum_k x_k \sum_i e_i \sum_j (\sigma_k^T\, M^*\, \sigma_k)_{i,j}\, x_j \\
&= \sum_k x_k \sum_i e_i\, (e_i^T \sigma_k^T\, M^*\, \sigma_k\, x) \\
&= \sum_i e_i\, e_i^T \sum_k \sigma_k^T\, M^*\, \sigma_k\, x_k\, x \\
&= \frac{1}{2}\, d\mathcal{M}_x\, x
\end{aligned}$$

□

We are now positioned to derive how $\mathcal{G}$ transforms in the coordinates corresponding to the Walsh basis $\mathcal{B}$. Whereas in the binary case ($c = 2$) real space suffices, the nonbinary case ($c > 2$) involves a transform matrix with complex entries. We must therefore move to complex space.

Transforming the representation

$$x \;=\; \sum x_j\, e_j$$

and then replacing x by $\widehat{x}$ in the resulting expression yields

$$x \;=\; \sum \widehat{x}_j^{\,C}\, \widehat{e_j}$$

The ability to pass between these two representations for x will be useful. By theorem 4.2 we can write $\mathcal{M}(x)$ as

$$\frac{1}{2}\, d\mathcal{M}_x x$$

Using the second representation for x given above and expanding by the bilinearity of $d\mathcal{M}_{(\cdot)}(\cdot)$ allows this to be written as

$$\frac{1}{2} \sum_{i,j} \widehat{x}_i^{\,C} \widehat{x}_j^{\,C} d\mathcal{M}_{\widehat{e_i}} \widehat{e_j}$$

Appealing to the first formula of theorem 4.1 and making the change of variables $i \oplus j = k$ leads to

$$\mathcal{M}(x) \;=\; \sqrt{n} \sum_k \widehat{e_k} \sum_i \widehat{x}_i^{\,C} \widehat{x}_{k\ominus i}^{\,C} \widehat{M}_{i, i\ominus k}$$

This derivation establishes

Theorem 4.3 *The k th component of $\mathcal{M}(x)$ with respect to the basis $\mathcal{B} = \{\widehat{e_0}, ..., \widehat{e_{n-1}}\}$ is*

$$\sqrt{n} \sum_i \widehat{x}_i^{\,C} \widehat{x}_{k\ominus i}^{\,C} \widehat{M}_{i, i\ominus k}$$

The next step is to calculate how $\mathcal{F}$ transforms in the coordinates corresponding to $\mathcal{B}$. Observe that

$$Fx \;=\; \sum (Fx)_j \, e_j \;=\; \sum \widehat{(Fx)}_j^{\,C} \, \widehat{e_j}$$

Therefore

$$\mathbf{1}^T F x \;=\; \sum_j \widehat{(Fx)}_j^{\,C} \, \mathbf{1}^T \widehat{e_j} \;=\; \sqrt{n} \sum_j \widehat{(Fx)}_j^{\,C} \, \widehat{e_0}^T \widehat{e_j} \;=\; \sqrt{n} \, \widehat{(Fx)}_0^{\,C}$$

This leads to

Theorem 4.4 *Let $F = diag(f)$ be the fitness matrix. The k th component of $\mathcal{F}(x)$ with respect to the basis $\mathcal{B}$ is*

$$\widehat{f}^{\,T} \sigma_k \, \widehat{x}^{\,C} \, / \, (\sqrt{n} \widehat{f}^{\,T} \widehat{x}^{\,C})$$

Sketch of proof: Since $\widehat{(Fx)}_j = e_j^T \widehat{F} \widehat{x}$, the discussion preceding theorem 4.4 shows it suffices that

$$e_j^T \widehat{F} \;=\; \widehat{f}^{\,H} \sigma_j / \sqrt{n}$$

This follows from theorem 3.1 by the following calculation,

$$\begin{aligned}
e_j^T \widehat{F} &= (e_j^H W^C)^C F \, W^C \\
&= (e_j^{H\wedge})^C F \, W^C \\
&= \widehat{e_j}^T \mathrm{diag}(f) \, W^C \\
&= f^T \mathrm{diag}(\widehat{e_j}) \, W^C \\
&= (f^H W^C)(W \mathrm{diag}(\widehat{e_j}) \, W^C) \\
&= \widehat{f}^H \mathrm{diag}\widehat{(e_{-j})} \\
&= \widehat{f}^H \sigma_j / \sqrt{n}
\end{aligned}$$

$\square$

Theorems 4.3 and 4.4 show that computing $\mathcal{G}$ in Fourier coordinates (i.e., computing with respect to the basis $\mathcal{B}$) is far more efficient than computing it in the standard basis. Consider the binary case ($c = 2$) for example. The i th component of $\mathcal{M}(x)$ involves the quadratic form

$$(\sigma_i x)^T M \sigma_i x$$

With positive mutation, M is dense and each computation is $O(n^2)$. Moreover, there are n such components to consider, giving $\mathcal{G}$ at least $O(n^3)$ complexity. In comparison, the cost of the corresponding component in Walsh coordinates is the size of Ω_k since the representation given by theorem 4.3 has nonzero terms only when $i^T(k \ominus i) = 0$ (this follows from theorem 3.2) which implies $i \in \Omega_k$ (in the general cardinality case, the number of nonzero terms is the same because

$i^T(k \ominus i) = 0$ implies that at every position i is either zero or matches k). Hence the total expense is bounded by the order of

$$\sum_k card(\Omega_k) \;=\; \sum_k 2^{\mathbf{1}^T k} \;=\; \sum_u 2^u \sum_k [\mathbf{1}^T k = u] \;=\; \sum_u 2^u \left(\begin{array}{c} \ell \\ u \end{array} \right) \;=\; n^{\log_2 3}$$

This compares favorably with $O(n^3)$. In Walsh coordinates, selection is the dominant cost. By theorem 4.4 it is $O(n^2)$, but this cost need not be incurred since transforming between coordinate systems via the fast Walsh transform (or fast Fourier transform in the general case) costs only $O(n \ln n)$ and selection can be computed in standard coordinates at a cost of $O(n)$.

There are reasons far more profound than computational efficiency, $O(n^{\log_2 3})$ vs $O(n^3)$, that the Fourier basis $\mathcal{B}$ appertains to the simple genetic algorithm. The next paper in this series will show how the Walsh basis can be applied to effectively triangulate the equations which correspond to mixing. A number of interesting consequences to having access to the triangularized system will be explored there.

This paper closes with the following section which is specialized to the binary case. The subject is how Walsh coordinates induce a decomposition of Λ into regions of space which are invariant under mixing.

5 Invariance

Before proving the invariance theorem, we discuss the relationship of mutation to the standard basis so as to provide a frame of reference.

If p is a population vector which does not have components in the direction of every basis vector, then p represents a population in which some string types are missing. In particular, string i is missing from the population exactly when $p_i = 0$. After mixing however, *every* string is expected to be represented, provided that mutation is positive, since any string has a nonzero probability of being produced by mutation and surviving crossover (being crossed with itself).

Hence there is no proper subset of the basis vectors whose linear span is invariant under $\mathcal{M}$. If some components of p were zero, those components become positive in the vector $\mathcal{M}(p)$. This is perhaps intuitive, since mutation "spreads out" the initial population to contain, in expectation, instances of every string type.

Nevertheless, in Walsh coordinates there are invariant subspaces – exponentially many of them – even when mutation is positive. That result is the subject of this section.

If S is a set of vectors, let $\mathcal{L}S$ denote their linear span. The invariance theorem is:

Theorem 5.1 *For all k, both $\mathcal{L}\{\widehat{e_i} : i \in \Omega_k\}$ and $\mathcal{L}\{\widehat{e_j} : j \notin \Omega_k\}$ are invariant under $\mathcal{M}$.*

Sketch of proof: We consider the second space first. Let $x = \sum \alpha_i \widehat{e_i}$ be an element of $\mathcal{L}\{\widehat{e_j} : j \notin \Omega_k\}$. Thus $i \in \Omega_k \Longrightarrow \alpha_i = 0$. By theorem 4.3, for such i

$$\widehat{e_i}^T \mathcal{M}(x) \quad = \quad 2^{\ell/2} \sum_{u \in \Omega_i} \alpha_u \, \alpha_{u \oplus i} \, \widehat{M}_{u, u \oplus i}$$

Since $\Omega_i \subset \Omega_k$, the coefficients α_u are zero.

Next let $x = \sum \alpha_i \widehat{e_i}$ be an element of $\mathcal{L}\{\widehat{e_i} : i \in \Omega_k\}$. For $j \notin \Omega_k$ we have as before,

$$\widehat{e_j}^T \mathcal{M}(x) \quad = \quad 2^{\ell/2} \sum_{u \in \Omega_j} \alpha_u \, \alpha_{u \oplus j} \, \widehat{M}_{u, u \oplus j}$$

where nonzero terms are subscripted by elments of Ω_j. Since $j \notin \Omega_k \Longrightarrow \alpha_j = 0$, every term will be zero provided that

$$u \in \Omega_j \quad \Longrightarrow \quad u \notin \Omega_k \ \vee \ u \oplus j \notin \Omega_k$$

This implication follows from the fact that $u \oplus (u \oplus j) = j \notin \Omega_k$. $\square$

Space has, for every choice of $k \in \Omega$, the orthogonal decomposition

$$\mathcal{L}\{\widehat{e_i} : i \in \Omega_k\} \times \mathcal{L}\{\widehat{e_j} : j \notin \Omega_k\}$$

Theorem 5.1 shows each factor space is invariant under mixing. Since mixing preserves Λ, the intersection of these spaces with Λ is also invariant. As a special case, each region is invariant under mutation.

Conclusion

This paper demonstrates a number of theoretical connections between the Walsh transform and the simple genetic algorithm. By working with the Fourier transform, of which the Walsh transform is a special case, we have carried out the investigation within a framework that extends directly to higher cardinality (nonbinary) alphabets.

A number of abstract theoretical results have been obtained which will be useful in supporting further inquiry into the basic nature and properties of the simple genetic algorithm and its associated mathematical objects. We briefly indicate below how the results of this paper are relevant in a wider context.

The major connections presented in this paper of the Walsh transform to the simple genetic algorithm are through:

1. Diagonalization of the linear group commuting with $\mathcal{M}$.
2. Simplification of M (a dense matrix becomes sparse).

3. Triangulation of M^* (giving access to its spectrum).

4. Explicit determination of the spectrum of $d\mathcal{M}$.

5. Complexity advantages of computations using the basis $\mathcal{B}$.

6. Orthogonal decompositions of space invariant under mixing.

The linear group $\{\sigma_k\}$ occurring in the definition of the mixing scheme identifies the Walsh transform as an inherent – not an arbitrary – object related to the simple genetic algorithm. This is the implication of the first connection enumerated above. The Walsh transform simultaneously diagonalizes this group. In other words, the columns of the Walsh matrix are related to fundamental symmetries which are embedded in the very definition of mixing; they are an orthonormal system of eigenvectors for the family $\{\sigma_k\}$.

The infinite population algorithm (i.e., $\mathcal{G}$) has recently been receiving increasing attention for various reasons. Not the least of these is the fact that the finite population model contains $\mathcal{G}$ as a defining component (see, for example, Juliany & Vose, 1994, and Vose, 1996). For both analysis and computation, significant progress has been achieved when the mathematical objects involved can be simplified. The second connection enumerated above is progress of that type. We have shown how the Fourier transform makes the fully positive (for nonzero mutation) matrix M sparse. Moreover, M is a fundamental component of both the infinite and finite population models as it occurs in the definition of $\mathcal{M}$ which is a composition factor of $\mathcal{G}$.

Getting a handle on the spectrum of M^* is of basic importance to better understanding $d\mathcal{M}$, and, as the sequel to this paper will demonstrate in complete detail, to determining the behavior of the mixing operator. The third connection enumerated above gives access to this spectrum. One of its concrete applications is the fourth connection; explicit determination of the spectrum of $d\mathcal{M}$. As shown in Koehler, Bhattacharyya & Vose (1998) and Vose & Wright (1995), this spectrum is related to the stability of fixed points of $\mathcal{G}$ (in fact, its connection to stability of fixed points lead to the discovery that minimal deceptive problems (Goldberg, 1987) are incomplete with respect to determining GA hard functions for the two bit infinite population model, see Juliany & Vose, (1994).

Working out how selection and mixing transform in the Fourier basis leads to the fifth connection enumerated above, providing the most efficient methods known to calculate with the infinite population model in the general case. The Fourier basis is crucial to decreasing the time involved in working with examples, to extending the size of simulations, and to conducting a wider range of computational experiments. The increase in efficiency provided is from $O(n^3)$ to $O(n^{1.585})$.

Interpreting the final connection, determining decompositions of space invariant under mixing, is beyond the scope of this paper. Suffice it to mention that Vose has worked out how it relates to quotient maps and the interpretation of genetic operators as taking place on equivalence classes (the connections, however, are in some respects complicated and would require more space than is feasible to include either here or in the sequel). The interested reader is referred to (Vose, 1998).

The sequel to this paper will apply basic theoretical results of this paper, and the Fourier basis in particular, to further demonstrate how the Fourier transform in general, and the Walsh transform in particular, appertains to the theory of the simple genetic algorithm. The focus there is on inverse problems and asymptotic behavior.

Acknowledgements

This research was supported by the National Science Foundation: IRI-8917545 and CDA-9115428. The second author was partially supported by a Montana NSF EPSCOR grant and a University of Montana University Research grant.

6 References

Aizawa, A. (1997). Fitness Landscape Characterization by Variance Decompositions, R. Belew & M. Vose (Eds.) *Foundations Of Genetic Algorithms, 4*, Morgan Kaufmann, 225-245.

Bethke, A. D. (1981). *Genetic Algorithms as Function Optimizers*, (Doctoral dissertation, University of Michigan). *Dissertation Abstracts International* 41(9), 3503B. (University Microfilms No. 8106101).

Birkhoff, G., & MacLane, S., (1953). *A Survey of Modern Algebra*, MacMillin, New York.

Deb, K., & Goldberg, D. (1993). Analyzing Deception in Trap Functions, in D. Whitley (Ed.), *Foundations of Genetic Algorithms II*, Morgan Kaufmann, 93-108.

Goldberg, D. (1987). Simple Genetic Algorithms and the Minimal, Deceptive Problem, in L. Davis (Ed.), *Genetic Algorithms and Simulated Annealing.* London: Pitman, 76-88.

Goldberg, D. (1989). Genetic Algorithms and Walsh Functions: Part II, Deception and Its Analysis, *Complex Systems*, *3*, 153-171.

Gantmacher, F. R. (1977). *Matrix Theory*, Chelsea, *1*, 290-291.

Goldberg, D. (1990). Construction of high-order deceptive functions using low-order Walsh coefficients, (IlliGAL Report No. 91010). Urbana: University of Illinois at Urbana-Champaign, Illinois Genetic Algorithms Laboratory.

Grefenstette, J. (1993). Deception Considered Harmful, in D. Whitely (Ed.) *Foundations of Genetic Algorithms II*, Morgan Kaufmann, 75-91.

Juliany J. & Vose, M. D. (1994). The Genetic Algorithm Fractal, *Evolutionary Computation, 2(2)*, 165-180.

Koehler, G. (1994). A Proof of the Vose-Liepins Conjecture, *Annals of Mathematics and Artificial Intelligence, 10*, 409-422.

Koehler, G. (1995). Diagonalizing the Simple GA Mixing Matrix, R. Belew & M. Vose (Eds.) *Foundations Of Genetic Algorithms, 4*, Morgan Kaufmann, 5-51.

Koehler, G., Bhattacharyya, S. & Vose, M. D. (1998). General Cardinality Genetic Algorithms, *Evolutionary Computation, 5(4)*, 439-459 .

Kok, J. N. & Floreen, P. (1995). Tracing the behavior of genetic algorithms using expected values of bit and walsh products, in L. J. Eshelman (ed.), *Proceedings of the sixth international conference on genetic algorithms,* Morgan Kaufmann Publishers, Inc., 201-208.

Vose, M. D. (1993). A Critical Examination Of The Schema Theorem, University of Tennessee, Technical Report (CS-93-212).

Vose, M. D. (1990). Formalizing Genetic Algorithms In the proceedings of an IEEE workshop: *Genetic Algorithms, Neural Nets, and Simulated Annealing Applied to Problems in Signal and Image Processing.* May, 1990, Glasgow, U.K.

Vose, M. D. (1996). Modeling Simple Genetic Algorithms, *Evolutionary Computation, 3(4)*, 453-472.

Vose, M. D. (1998). *The Simple Genetic Algorithm: Foundations and Theory*, MIT Press, In Press.

Vose, M. D. & Liepins, G. (1991). Punctuated Equilibria In Genetic Search, *Complex Systems, 5*, 31-44

Vose, M. D. & Wright, A. H. (1994). Simple Genetic Algorithms with Linear Fitness, *Evolutionary Computation, 2(4)*, 347-368.

Vose, M. D. & Wright, A. H. (1995). Stability of Vertex Fixed Points and Applications, in L. D. Whitley & M. D. Vose (Eds.) *Foundations Of Genetic Algorithms III*, Morgan Kaufmann, 103-113.

Vose, M. D. & Wright, A. H. (1996). The Walsh Transform and the Theory of the Simple Genetic Algorithm, In S. Pal & P. Wang (Eds), *Genetic Algorithms for Pattern Recognition*, CRC Press, 25-44.

Weinberger, E. D. (1991). Fourier and Taylor Series on fitness landscapes, *Biological Cybernetics, 65*, 321-330.