

---

# Structural Search Spaces and Genetic Operators

**Jonathan E. Rowe**

School of Computer Science, University of Birmingham, Birmingham B15 2TT, Great Britain

J.E.Rowe@cs.bham.ac.uk

**Michael D. Vose**

Computer Science Department, University of Tennessee, Knoxville, TN 37996, USA

vose@cs.utk.edu

**Alden H. Wright\***

Dept. of Computer Science, University of Montana, Missoula, Montana 59812, USA

wright@cs.umt.edu

---

## Abstract

In a previous paper (Rowe et al., 2002), aspects of the theory of genetic algorithms were generalised to the case where the search space,  $\Omega$ , had an arbitrary group action defined on it. Conditions under which genetic operators respect certain subsets of  $\Omega$  were identified, leading to a generalisation of the term *schema*. In this paper, search space groups with more detailed structure are examined. We define the class of *structural crossover* operators that respect certain schemata in these groups, which leads to a generalised schema theorem. Recent results concerning the Fourier (or Walsh) transform are generalised. In particular, it is shown that the matrix group representing  $\Omega$  can be simultaneously diagonalised if and only if  $\Omega$  is Abelian. Some results concerning structural crossover and mutation are given for this case.

## Keywords

Genetic Algorithms, Mixing Matrix, Schema, Group Action, Crossover, Mutation, Walsh Transform

## 1 Introduction

This paper is a sequel to the paper “Group properties of crossover and mutation” (Rowe et al., 2002). In that paper, a number of the elements of genetic algorithm theory were generalised. It turns out that a key factor in relating the representation chosen for a problem to the design of operators (crossover and mutation) is the symmetries inherent in that representation, and the way in which the operators respect them. Such symmetries are captured mathematically in terms of a group of permutations acting on the underlying search space. One of the main considerations of the previous paper was to determine conditions under which operators *commute* with such a group action. The importance of this is that operators which do commute can be expressed in terms of a *mixing matrix* together with a set of permutation matrices, dramatically simplifying the equations of evolution. Another consideration of that paper was to find conditions under which certain subsets of the search space are *respected* by crossover (that is, offspring will inherit such properties from parents). This led to a natural generalisation of the term *schema* in terms of subgroup structure. In section 2 we briefly review some of the more important results.

---

\* Alden Wright completed some of this work while visiting the University of Birmingham, supported by EPSRC grant GR/R47394.

In the current paper, we will concentrate on a special case. Firstly, we will be assuming that the search space itself forms a group with some appropriately defined operator (as is the case for fixed-length binary strings, for example, under bitwise exclusive-or). Secondly, we will consider groups that can be expressed as direct sums of normal subgroups. Put simply, this means we are considering search spaces which are made up of a number of independent component subspaces. For example, with fixed-length binary strings of length  $\ell$ , we have  $\ell$  components, each of which may be identified with the group  $\mathcal{Z}_2$ . Our generalization leads us to consider strings whose components range over arbitrary alphabets. In addition, we consider search spaces that have other group structures. One example would be the problem of arranging schedules for a number of machines: a particular solution would comprise a list of permutations, one for each machine. Such search spaces, along with the types of crossover and mutation which naturally act on them, are called *structural*. They are defined and examined in section 3.

Section 4 then shows how structural crossover and mutation operators naturally project onto families of competing schemata. We call this result the *Schemata Projection Theorem* since it describes the effect of mixing on all schemata simultaneously. Section 5 deals with generalising the Walsh transform to structural search spaces. The Walsh basis is a natural setting for genetic algorithm theory when dealing with fixed-length binary strings. We would like to know if, and under what conditions, a similar basis exists for other structural search spaces. It turns out that there is indeed such a basis, given by the Fourier transform, exactly for those search spaces which are Abelian (that is, commutative as groups). For non-Abelian groups (for example, when the travelling salesman problem is identified with the group of permutations of the set of cities) there can be no equivalent of the Walsh transform. In the case when the group is Abelian, there is considerably more algebraic structure that can be exploited. Some of the consequences of this are worked out in section 6. There are thus three main contributions in this paper:

- Structural search spaces and operators are defined, and their properties explored. These are natural generalisations of the standard GA acting on binary strings with mask-based crossover and mutation. Section 3 shows how this family of GAs fits within the general framework established in our earlier paper.
- The Schemata Projection Theorem shows how structural crossover and mutation project onto subspaces of the search space defined by families of competing schemata. Section 4 describes this result and discusses its relationship to the Schema Theorem.
- The Walsh basis is generalised, and exact conditions are found for which a search space (with appropriate group structure) will have such a basis. The key result of section 5 is that a Walsh-like transform only exists for search spaces comprising strings over finite alphabets (with possibly varying cardinality).

The motivation for this work is as follows. Suppose you are presented with a new combinatorial optimisation problem and you wish to design a genetic algorithm to tackle it. A key decision that has to be made is how to represent the candidate solutions, and how to design genetic operators (crossover and mutation) to act upon them. It is by no means obvious how to do this in a systematic way, if there is no direct mapping onto binary (or higher cardinality) strings. Suppose, however, that you understand enough about the structure of the problem that you can identify certain inherent symmetries,

and certain related properties that you consider significant (so that you would wish offspring to inherit them from parents). If you can characterise these symmetries by defining a group of permutations that acts transitively on the search space, and if you can identify your important properties with the subgroup structure of this group (they correspond, if you will, to the natural “folds” in the search space—see (Vose, 1991)), then our theory applies and will tell you something about the kinds of crossover and mutation operators that can be defined on your problem. That is the subject of our previous paper. If you are lucky enough to find that your problem can be broken down into some “independent” subspaces, then even stronger results hold. That is the subject of the current paper. Where might these symmetries come from? We speculate that they can be formally derived from any natural neighbourhood structure which you impose on your search space. In other words, they are the symmetries of the underlying landscape graph. This is the subject of a future paper.

### Notation

In order to facilitate the reading of this paper, we have adopted the following conventions:

$\Omega$	the search space	$n$	the cardinality of $\Omega$
$u, v, w$	elements of $\Omega$	$\pi, \rho, \nu$	permutations of $\Omega$
$p, q$	population vectors	$\mathcal{B}$	the set of binary masks
$a, b$	elements of $\mathcal{B}$	$i, j, k$	indices of elements of $\Omega$ and $\mathcal{B}$

The notation  $[\dots]$  denotes 1 if “ $\dots$ ” is true, and 0 otherwise.

## 2 Previous results

### 2.1 Search space symmetries, crossover and mutation

In a previous paper (Rowe et al., 2002), aspects of the theory of genetic algorithms were generalised to the case where the search space  $\Omega$  is acted on by an arbitrary group  $(L, \circ)$ . That is,  $L$  is a set of permutations of  $\Omega$  which forms a group under function composition. The action of  $\pi \in L$  on some element  $w \in \Omega$  is denoted by  $\pi(w)$ . The group action, denoted by  $L(\Omega)$ , is assumed to be *reduced*. That is, for every  $u, v \in \Omega$  there exists  $\pi \in L$  such that  $\pi(u) = v$  (so that  $L(\Omega)$  acts transitively), and there is only one group element, namely the identity, which is also the identity map on  $\Omega$ .<sup>1</sup>

For each  $\pi \in L$ , define the  $n \times n$  permutation matrix

$$(\sigma_\pi)_{u,v} = [u = \pi(v)]$$

The indices  $u$  and  $v$  run through the set  $\Omega$ . The set of all such permutation matrices forms a group (under matrix multiplication) isomorphic to  $L$ . Each permutation matrix can also be thought of as a linear map  $\sigma_\pi : \mathbb{R}^n \rightarrow \mathbb{R}^n$ .

The underlying mixing scheme  $\mathcal{M}$  of the genetic algorithm which describes the effects of crossover and mutation on a population is a map of the simplex to itself

$$\mathcal{M} : \Lambda \longrightarrow \Lambda$$

where  $\Lambda = \{p \in \mathbb{R}^n : p_w \geq 0, \sum p_w = 1\}$  is the completion of the state space of finite populations. The effect of crossover alone is given by the crossover scheme  $\mathcal{C}$  (obtained

<sup>1</sup>We can always make this assumption simply by identifying group elements that have the same effects on  $\Omega$ .

from  $\mathcal{M}$  by choosing mutation rate zero), and the effect of mutation alone is given by the mutation scheme  $\mathcal{U}$  (obtained from  $\mathcal{M}$  by choosing crossover rate zero). The framework is so constructed that the intuitive result  $\mathcal{M} = \mathcal{C} \circ \mathcal{U}$  holds, where  $\circ$  denotes function composition.<sup>2</sup>

A map  $g : \Lambda \rightarrow \Lambda$  is said to *commute* with  $L(\Omega)$  if  $g \circ \sigma_\pi = \sigma_\pi \circ g$  for all  $\pi \in L$ . The following two results (Rowe et al., 2002) were proved concerning conditions under which crossover and mutation (and hence mixing in general) commute with  $L(\Omega)$ .

**Result 1 (Rowe et al., 2002), Theorem 5** *Let  $s(u, v, w) = (r(u, v, w) + r(v, u, w))/2$  where  $r(u, v, w)$  is the probability that crossover produces  $w$  from parents  $u$  and  $v$ . The crossover scheme commutes with  $L(\Omega)$  if and only if*

$$s(\pi(u), \pi(v), \pi(w)) = s(u, v, w)$$

for all  $u, v, w \in \Omega$  and  $\pi \in L$ .

**Result 2 (Rowe et al., 2002), Theorem 6** *The mutation scheme is a linear operator. Let  $U$  denote its matrix. The mutation scheme commutes with  $L(\Omega)$  if and only if*

$$U_{\pi(u), \pi(v)} = U_{u, v}$$

for all  $u, v \in \Omega$  and  $\pi \in L$ .

Since  $\Omega$  is finite, it may be enumerated. Let 0 refer to the 0th element of  $\Omega$  (with respect to some enumeration). When mixing commutes with  $L(\Omega)$ , the mixing scheme can be written in terms of a single mixing matrix  $M$  together with the associated set of permutation matrices:

$$\mathcal{M}(\mathbf{p})_w = \mathbf{p}^T (\sigma_\pi M \sigma_\pi^T) \mathbf{p}$$

where  $\pi \in L$  is chosen such that  $\pi(0) = w$  and where  $M_{u, v} = s(u, v, 0)$  is the probability that parents  $u$  and  $v$  produce offspring 0 after crossover and mutation.<sup>3</sup> We can recover the probability that some arbitrary offspring  $w$  is produced from  $s(u, v, w) = s(u, v, \pi(0)) = s(\pi^{-1}(u), \pi^{-1}(v), 0) = M_{\pi^{-1}(u), \pi^{-1}(v)}$ .

## 2.2 Schemata and subgroup structure

The following definition generalises the notion of *schemata*.

**Definition 1** *Let  $\mathcal{N}$  be a set of subgroups of  $L$ . For each  $A \in \mathcal{N}$  and each  $\pi \in L$ , the set*

$$\pi \circ A(0) = \{w \in \Omega \mid w = (\pi \circ \nu)(0) \text{ for some } \nu \in A\}$$

is called a  $\mathcal{N}$ -schema. When the context is not ambiguous,  $\mathcal{N}$ -schemata are referred to simply as schemata.

**Definition 2** *If  $u, v$  are contained in a schema then  $u$  and  $v$  are said to share that schema. They are said to be compatible (with respect to  $\mathcal{N}$ ) if whenever they share a  $\mathcal{N}$ -schema, it is also shared by the element  $0 \in \Omega$ .*

<sup>2</sup>The seemingly ambiguous use of  $\circ$  (it represents both composition and the group operation of  $L$ ) is resolved by context.

<sup>3</sup>Notice that crossover and mutation together combine to form a single quadratic operator (see Theorem 1 of (Rowe et al., 2002)).

These definitions will be looked at more closely in section 3.

It may appear that the choice of the element  $0 \in \Omega$  will affect which sets become schemata. However, this is not the case. As previously alluded in (Rowe et al., 2002),  $\mathcal{N}$ -schemata are *natural* (i.e., independent of the enumeration of  $\Omega$  which determines  $0 \in \Omega$ ) when  $\mathcal{N}$  contains normal subgroups, since in that case the  $\mathcal{N}$ -schemata are orbits.

However, even when  $L$  is Abelian, the concept of “compatibility” need not be natural. For instance, consider  $L = \Omega = \mathcal{Z}_2 \times \mathcal{Z}_2$ , where  $\mathcal{Z}_2$  is the integers modulo 2 under addition and  $\Omega$  acts on itself by the group operation. With respect to  $\mathcal{N} = \{\{(0, 0), (1, 0)\}\}$  and the enumeration  $(0, 0), (0, 1), (1, 0), (1, 1)$ , the elements  $(0, 0)$  and  $(1, 0)$  are compatible. But they are not compatible with respect to the enumeration  $(0, 1), (1, 1), (0, 0), (1, 0)$ .

**Definition 3** *Crossover is said to respect a subset of  $\Omega$  if crossing two elements of that subset always creates an offspring in that subset (Radcliffe, 1992).*

The following result (Rowe et al., 2002) tells us exactly when crossover respects schemata.

**Result 3 (Rowe et al., 2002), Theorem 13** *Suppose crossover commutes with  $L(\Omega)$  and has mixing matrix  $M$ . Let  $\mathcal{N}$  be a collection of subgroups of  $L$ . Then crossover respects all  $\mathcal{N}$ -schemata if and only if*

$$M_{u,v} > 0 \implies u, v \text{ are compatible}$$

For the remainder of this paper we will consider the special case where  $\Omega$  has a group structure and acts on itself (thus  $L = \Omega$ , and to simplify exposition,  $L(\Omega)$  will be abbreviated by  $\Omega$ ). We will be particularly interested in the subgroup structure of  $\Omega$  in this case, and in genetic operators that respect that structure. From this point forward assume the following. The action of  $(\Omega, \oplus)$  on itself is defined by

$$u(v) = u \oplus v$$

Let  $\ominus v$  denote the inverse of  $v$ , and abbreviate  $u \oplus (\ominus v)$  by  $u \ominus v$ . The identity element of  $\Omega$  is denoted by  $0$ , or to put it another way, the enumeration is chosen such that the 0th element is the identity. It follows that  $\mathcal{N}$ -schemata are cosets (that is, they have the form  $w \oplus A$  for some  $w \in \Omega$  and  $A \in \mathcal{N}$ ).

### 2.3 A note on the definition of group action

Notice that when  $\Omega$  itself is a group, it has a dual interpretation. It is both the underlying set (which is the search space) and a set of permutations (bijections) which act on that underlying set (via the definition  $u(v) = u \oplus v$  described above). In (Rowe et al., 2002) we considered the question of when it might be possible to give the search space a group structure that was compatible with the group  $L(\Omega)$  that acts upon it. Theorem 17 of that paper tells us to consider another set of permutations of  $\Omega$ . This set of permutations  $R(\Omega)$  is made up of all bijections  $\rho : \Omega \rightarrow \Omega$  that have the property

$$\pi \in L(\Omega) \implies \pi \circ \rho(w) = \rho \circ \pi(w)$$

for all  $w \in \Omega$ . That is, the permutations in  $R$  commute with those in  $L$ , and in fact this set forms a group (under function composition). Theorem 17 then tells us that if  $\Omega$  can

be given a compatible group structure, then  $L$  is anti-isomorphic to  $R$ . Moreover, just as  $L$  gives us a group structure on  $\Omega$ , so does  $R$  — it is the *reverse* of that given by  $L$ .

In our current situation we have  $L = \Omega$ , and so obviously  $\Omega$  has a group structure compatible with  $L$ . Theorem 17 tells us that the set of permutations  $R$  forms a group anti-isomorphic to  $\Omega$ . That is, there exists a bijection  $\phi : R \rightarrow \Omega$  such that

$$\phi(\pi \circ \rho) = \phi(\rho) \oplus \phi(\pi)$$

Moreover, we could use the structure of  $R$  to define a group structure on  $\Omega$ , and that this is the reverse of the action defined by  $L$ . Under this alternative group structure we would have

$$\phi(u)(v) = v \oplus u$$

It is clear that this new action commutes with that defined by  $L$  above since

$$\phi(u) \circ v(w) = \phi(u)(v \oplus w) = v \oplus w \oplus u = v \circ \phi(u)(w)$$

Consequently, the whole of the theory that follows in this paper could have been constructed with this reverse definition of group action based on  $R$ . Every result would have gone through in a symmetrical fashion.

### 3 Structural genetic operators

#### 3.1 Search spaces

We generalize (Vose, 1999) by introducing a class of genetic operators associated with a certain subgroup structure. Suppose  $\Omega$  has nontrivial subgroups  $A_0, \dots, A_{\ell-1}$  such that for all  $i, j \in \{0, 1, \dots, \ell - 1\}$  and  $w \in \Omega$ ,

1.  $\Omega = A_0 \oplus \dots \oplus A_{\ell-1}$
2.  $i \neq j \implies A_i \cap A_j = \{0\}$
3.  $w \oplus A_i = A_i \oplus w$

then  $\Omega$  is the *internal direct sum* of the  $A_i$  (which are *normal* subgroups of  $\Omega$ ) and each element  $w \in \Omega$  has a unique representation  $w = w_0 \oplus \dots \oplus w_{\ell-1}$  where  $w_i \in A_i$ . The map

$$w \longmapsto \langle w_0, \dots, w_{\ell-1} \rangle$$

is an isomorphism between  $\Omega$  and the product group  $A_0 \times \dots \times A_{\ell-1}$  (see (Lang, 1993)), where

$$\langle u_0, \dots, u_{\ell-1} \rangle \oplus \langle v_0, \dots, v_{\ell-1} \rangle = \langle u_0 \oplus v_0, \dots, u_{\ell-1} \oplus v_{\ell-1} \rangle$$

and

$$\ominus \langle w_0, \dots, w_{\ell-1} \rangle = \langle \ominus w_0, \dots, \ominus w_{\ell-1} \rangle$$

**Definition 4** Let  $(\Omega, \oplus)$  be a finite group which can be written as the internal direct sum of normal subgroups. Then  $\Omega$  considered as a search space is called structural.

We will assume for the remainder of this paper that  $\Omega$  is structural, and identify  $w \in \Omega$  with  $\langle w_0, \dots, w_{\ell-1} \rangle$ .

### 3.2 Examples

**Strings over arbitrary alphabets** As an example, consider a length  $\ell$  string representation where the cardinality of the alphabet at string position  $j$  is  $c_j$ . The alphabet at position  $j$  can be identified with  $\mathcal{Z}_{c_j}$  (the integers modulo  $c_j$ ), and the  $\oplus$  operator can be componentwise addition (modulo  $c_j$  at position  $j$ ). Then  $\Omega$  is isomorphic to the direct product  $\mathcal{Z}_{c_0} \times \dots \times \mathcal{Z}_{c_{\ell-1}}$ . This example extends the situation considered in (Koehler et al., 1997) where  $c_0 = c_1 = \dots = c_{\ell-1}$ . The standard example of fixed-length binary strings is a special case in which  $c_j = 2$  for all positions  $j$ .

A concrete example of the above is:  $\ell = 2$ ,  $c_0 = 3$ ,  $c_1 = 2$ , so  $\Omega$  is isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ . When we write elements of  $\Omega$  as strings, the standard practice of putting the least significant bit to the right is followed. Thus,

$$\Omega = \{00, 01, 10, 11, 20, 21\} = \{0, 1, 2, 3, 4, 5\}$$

The group operator works by applying addition modulo 3 to the left bit, and addition modulo 2 to the right bit. For example  $21 \oplus 11 = 00$ . The element 00 is the identity.

**Scheduling jobs on machines** Suppose a factory process involves  $\ell$  machines and each machine has a number of jobs to perform. Let  $m_j$  be the number of jobs that machine  $j$  has to perform. The problem is to come up with a schedule for each machine. That is, for each machine  $j$ , we have to specify an ordering of the jobs  $1, 2, \dots, m_j$ . A particular schedule therefore comprises  $\ell$  permutations.

To take a specific example, suppose there are three machines, and each machine has three jobs, which we will number 1, 2 and 3. A particular schedule might be denoted  $(123, 321, 132)$  which indicates that the first machine performs its tasks in the order 1, 2, 3, the second in the order 3, 2, 1 and the third in the order 1, 3, 2.

The set of permutations on  $m$  items forms the group  $S_m$  (under composition of permutations). Our example therefore corresponds to the group  $S_3 \times S_3 \times S_3$ . The group operator  $\oplus$  applies permutation composition to each component independently. For example:  $(123, 321, 132) \oplus (123, 312, 231) = (123, 132, 321)$ . The identity element is  $(123, 123, 123)$ . Note that this group is not Abelian (that is,  $u \oplus v \neq v \oplus u$ ).

### 3.3 Binary masks

The set  $\mathcal{B}$  of *binary masks* corresponding to  $\Omega$  is

$$\mathcal{B} = \{\langle b_0, \dots, b_{\ell-1} \rangle : b_i \in \mathcal{Z}_2\}$$

where  $\mathcal{Z}_2$  is the set of integers modulo 2. Note that  $\mathcal{B}$  is an Abelian group under component-wise addition modulo 2. It is notationally convenient to let  $\oplus$  also denote the group operation on  $\mathcal{B}$ ; hence  $\oplus$  is *polymorphic*.<sup>4</sup> Let  $\otimes$  denote component-wise multiplication on  $\mathcal{B}$ , and let  $\mathbf{1} \in \mathcal{B}$  be the identity element for  $\otimes$ . For  $b \in \mathcal{B}$ , define  $\bar{b}$  by

$$\bar{b} = \mathbf{1} \oplus b$$

It is notationally convenient to extend  $\otimes$  to a commutative operator acting also between elements  $b \in \mathcal{B}$  and elements  $w \in \Omega$  by

$$\langle b_0, \dots, b_{\ell-1} \rangle \otimes \langle w_0, \dots, w_{\ell-1} \rangle = \langle b_0 w_0, \dots, b_{\ell-1} w_{\ell-1} \rangle$$

<sup>4</sup>An operator is polymorphic when its definition depends upon the type of its arguments.

where  $0w_i = 0 \in \Omega$  and  $1w_i = w_i \in \Omega$ . Here the right hand sides are elements of  $\Omega$ ; hence  $\otimes$  is polymorphic. To simplify notation,  $\otimes$  takes precedence over  $\oplus$  by convention. If  $b \in \mathcal{B}$  is a mask then  $\#b$  denotes the number of ones it contains. Some useful algebraic identities concerning the properties of masks and group elements can be found in the appendix.

### 3.4 Structural crossover

Let  $\chi$  be a probability distribution over the set of binary masks,

$$\chi_b = \text{the probability of mask } b$$

**Definition 5** Structural crossover with distribution  $\chi$  applied to parents  $u$  and  $v$  corresponds to choosing binary mask  $b$  with probability  $\chi_b$  and then producing the offspring  $u \otimes b \oplus \bar{b} \otimes v$ .

The probability that parents  $u, v$  have child  $w$  under structural crossover is therefore

$$r(u, v, w) = \sum_{b \in \mathcal{B}} \chi_b [u \otimes b \oplus \bar{b} \otimes v = w]$$

The corresponding crossover scheme  $\mathcal{C}$  is also called structural and satisfies

$$\mathcal{C}(p)_w = \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = w]$$

where  $p$  is a population vector and  $u, v \in \Omega$ . For example, for uniform crossover with crossover rate  $c$ , the probability distribution  $\chi$  is given by  $\chi_0 = 1 - c + c/2^\ell$  and  $\chi_b = c/2^\ell$  for  $b \neq 0$ .

It is notable that this formalization is syntactically identical to that given in (Vose, 1999), and the algebraic properties of  $\oplus$ ,  $\otimes$ , and  $\bar{\phantom{x}}$  are nearly the same. This makes possible the application of proof techniques used there to establish the results of this paper.

**Theorem 1** Structural crossover commutes with  $\Omega$ .

**Proof** Let  $u, v, w, z \in \Omega$ , and  $b \in \mathcal{B}$ . Using the properties of  $\oplus$  and  $\otimes$  given in the appendix,

$$\begin{aligned} [(z \oplus u) \otimes b \oplus \bar{b} \otimes (z \oplus v) = z \oplus w] &= [z \otimes b \oplus u \otimes b \oplus \bar{b} \otimes u \oplus \bar{b} \otimes v = z \oplus w] \\ &= [z \otimes b \oplus z \otimes \bar{b} \oplus u \otimes b \oplus \bar{b} \otimes v = z \oplus w] = [z \oplus u \otimes b \oplus \bar{b} \otimes v = z \oplus w] \\ &= [u \otimes b \oplus \bar{b} \otimes v = w] \end{aligned}$$

It follows that  $r(z \oplus u, z \oplus v, z \oplus w) = r(u, v, w)$ . Applying result 1 completes the proof.  $\square$

**Definition 6** Given binary mask  $b$ , define the subgroup  $\Omega_b$  by  $\Omega_b = b \otimes \Omega$

From now on we will let  $\mathcal{N}$  be the collection of all such subgroups,  $\mathcal{N} = \{\Omega_b : b \in \mathcal{B}\}$

### 3.5 Examples

**Strings over arbitrary alphabets** Continuing the example above ( $\ell = 2$  and  $\Omega$  isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ ), we have  $\Omega = \{00, 01, 10, 11, 20, 21\}$ .  $\mathcal{N}$  consists of four subgroups,  $\Omega_{00} = \{00\}$ ,  $\Omega_{01} = \{00, 01\}$ ,  $\Omega_{10} = \{00, 10, 20\}$ , and  $\Omega_{11} = \Omega = \{00, 01, 10, 11, 20, 21\}$ . There are three  $\mathcal{N}$ -schemata which are cosets of  $\Omega_{01}$ , namely



$\Omega_{01} = \{00, 01\}$ ,  $\Omega_{01} \oplus 10 = \{10, 11\}$ , and  $\Omega_{01} \oplus 20 = \{20, 21\}$ . These correspond to the traditional schemata  $0*$ ,  $1*$  and  $2*$  respectively. There are two  $\mathcal{N}$ -schemata which are cosets of  $\Omega_{10}$ , namely  $\Omega_{10} = \{00, 10, 20\}$  and  $\Omega_{10} \oplus 01 = \{01, 11, 21\}$  corresponding to  $*0$  and  $*1$ . The cosets of  $\Omega_{00}$  consist of the schemata defined by singleton sets, and  $\Omega_{11}$  corresponds to the whole search space, given by  $**$ .

**Scheduling jobs on machines** In this example  $\Omega$  is isomorphic to  $S_3 \times S_3 \times S_3$ . The set of binary masks is  $\{000, 001, 010, 011, 100, 101, 110, 111\}$  and each gives rise to a normal subgroup. For example taking the mask 011 we obtain the subgroup corresponding to schedules for the second and third machines, with the schedule for the first machine fixed at the identity (123). Its six cosets give rise to the schemata

$$(123, *, *), (132, *, *), (213, *, *), (231, *, *), (312, *, *), (321, *, *)$$

where the symbol  $*$  indicates that any permutation may be placed in the corresponding position. Thus each of these schemata comprises 36 possible schedules.

### 3.6 Properties of structural crossover

We now prove two important properties of structural crossover, namely that all the  $\mathcal{N}$ -schemata are respected, and that it is *pure* (that is, crossing a parent with itself is guaranteed to always clone that parent). We begin with a technical theorem about the structure of the  $\mathcal{N}$ -schemata.

**Theorem 2** *The  $\mathcal{N}$ -schemata are separating.*<sup>5</sup>

**Proof** It suffices to show that the  $\Omega_b$  are normal subgroups with trivial intersection (see the discussion following Definition 6 in (Rowe et al., 2002)). The intersection is  $\{0\}$  since  $\{0\} \in \mathcal{N}$ . The  $\Omega_b$  are normal since, for any  $w \in \Omega$

$$\begin{aligned} w \oplus \Omega_b &= w \otimes \bar{b} \oplus w \otimes b \oplus b \otimes \Omega = b \otimes (w \oplus \Omega) \oplus w \otimes \bar{b} = b \otimes (\Omega \oplus w) \oplus w \otimes \bar{b} \\ &= b \otimes \Omega \oplus w \otimes b \oplus w \otimes \bar{b} = \Omega_b \oplus w \end{aligned}$$

□

**Theorem 3** *Let  $\mathcal{C}$  be a structural crossover scheme. Then  $\mathcal{C}$  respects all  $\mathcal{N}$ -schemata.*

**Proof** Let  $\mathcal{C}$  have mixing matrix  $M$ . By result 3, it suffices to show that if  $M_{u,v} > 0$  and  $u \ominus v \in \Omega_b$  (that is,  $u$  and  $v$  share a schema) then

$$\Omega_b \oplus u = \Omega_b \oplus v = \Omega_b$$

This would follow from  $v \in \Omega_b$ . It therefore suffices to show  $\ominus v \in \Omega_b$  (since this is a subgroup). Now, since  $u \ominus v \in \Omega_b$ , there is some  $w$  for which  $u = b \otimes w \oplus v$ . Hence,

$$\begin{aligned} M_{u,v} &= \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} [(b \otimes w \oplus v) \otimes a \oplus \bar{a} \otimes v = 0] = \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} [b \otimes (w \otimes a) \oplus v = 0] \\ &= \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} [b \otimes (w \otimes a) = \ominus v] \end{aligned}$$

Since  $M_{u,v} > 0$  there is some nonzero term in the sum above, i.e., some  $a$  for which  $\ominus v = b \otimes (w \otimes a)$ . Therefore  $\ominus v \in \Omega_b$  as required. □

<sup>5</sup>The reader is referred to (Rowe et al., 2002) for the general technical definition. In this paper, the first sentence of the proof of theorem 2 adequately describes the concept.

Recall that a crossover operator is pure if and only if the result of crossing any  $w \in \Omega$  with itself is always  $w$ .

Recall that schemata  $H, H'$  are complementary if they can be written as  $H = w \oplus A$  and  $H' = w \oplus A'$  where  $A, A' \in \mathcal{N}$  and the  $\{A, A'\}$ -schemata are separating. Note that for all  $w \in \Omega$  and  $b \in \mathcal{B}$ , schemata  $\Omega_b \oplus w$  and  $\Omega_{\bar{b}} \oplus w$  are complementary.

**Theorem 4** *Let  $\mathcal{C}$  be a structural crossover scheme with mixing matrix  $M$ . Then  $\mathcal{C}$  is pure,  $M_{w,w} = [w = 0]$ , and  $0$  is the only element compatible with itself.*

**Proof** This can be verified by direct computation. However, it is a direct consequence of more general considerations: theorems 1, 2, 3 of this paper and theorems 14, 15 of (Rowe et al., 2002).  $\square$

Compatibility issues (see definition 2) are important in view of theorems 1, 3, and result 3. The next theorem shows how they may easily be dealt with. Define the operator  $\odot$  to be an associative and commutative operator acting between elements  $u, v \in \Omega$  by

$$(u \odot v)_k = [u_k = v_k]u_k$$

That is,  $u \odot v$  agrees with  $u$  and  $v$  in components in which they themselves agree, and is the identity elsewhere.

**Theorem 5** *Elements  $u, v \in \Omega$  are compatible if and only if  $u \odot v = 0$ .*

**Proof** Suppose  $u, v$  are compatible and  $u_k = v_k$  for some index  $k$ . Define  $b \in \mathcal{B}$  by  $b_i = [i = k]$ . Note that  $u \otimes b = v \otimes b$ . Since

$$\begin{aligned} u &= u \otimes b \oplus u \otimes \bar{b} \in u \otimes b \oplus \Omega_{\bar{b}} \\ v &= v \otimes b \oplus v \otimes \bar{b} \in v \otimes b \oplus \Omega_{\bar{b}} \end{aligned}$$

it follows that  $u, v \in u \otimes b \oplus \Omega_{\bar{b}}$ . By compatibility,  $0 \in u \otimes b \oplus \Omega_{\bar{b}}$ , and operating by  $\otimes b$  yields

$$0 = 0 \otimes b \in u \otimes b \oplus b \otimes \Omega_{\bar{b}} = \{u \otimes b\}$$

Hence  $u_k = 0$  and  $(u \odot v)_k = [u_k = v_k]u_k = 0$ . Moreover, if  $u_k \neq v_k$ , then by definition  $(u \odot v)_k = 0$ .

Conversely, suppose  $u \odot v = 0$  and suppose  $u, v \in w \oplus \Omega_{\bar{b}}$  for some  $w \in \Omega$  and  $b \in \mathcal{B}$ . Since

$$w \oplus \Omega_{\bar{b}} = w \otimes b \oplus w \otimes \bar{b} \oplus \Omega_{\bar{b}} = w \otimes b \oplus \Omega_{\bar{b}}$$

operating by  $\otimes b$  yields

$$u \otimes b, v \otimes b \in w \otimes b \oplus b \otimes \Omega_{\bar{b}} = \{w \otimes b\}$$

Suppose  $k$  is such that  $b_k = 1$ . It follows from the containment above that  $u_k = v_k = w_k$ , and since  $0 = (u \odot v)_k = [u_k = v_k]u_k$ , it follows that  $w_k = 0$ . Therefore  $w \otimes b = 0$ . Therefore  $w = w \otimes \bar{b}$  and so  $w \in \Omega_{\bar{b}}$ . In other words,  $w \oplus \Omega_{\bar{b}} = \Omega_{\bar{b}}$ .  $\square$

### 3.7 Examples

**Strings over arbitrary alphabets** Structural crossover on strings corresponds to the usual crossover by masks. Thus we obtain standard one-point, two-point and uniform crossover, simply by picking appropriate probability distributions over the set of masks. For example, when  $\Omega = \mathcal{Z}_3 \times \mathcal{Z}_2$  we obtain one-point crossover (with crossover probability of 1/3) by the distribution:  $\chi_{00} = 1/3, \chi_{01} = 0, \chi_{10} = 1/3,$

$\chi_{11} = 1/3$ . So taking strings  $u = 11$  and  $v = 20$  and applying the mask  $b = 10$  yields the offspring

$$u \otimes b \oplus \bar{b} \otimes v = (11 \otimes 10) \oplus (01 \otimes 20) = 10 \oplus 00 = 10$$

that is, we get the first component of  $u$  and the second component of  $v$ .<sup>6</sup>

Schemata are respected by this kind of crossover. For example, the strings 20 and 21 are both elements of the schema  $2*$  and so we are guaranteed that any offspring of these two strings will also be in this schema. Moreover, structural crossover is pure, as illustrated by crossing the string 21 with itself, using mask  $b = 10$ ,

$$(21 \otimes 10) \oplus (01 \otimes 21) = 20 \oplus 01 = 21$$

**Scheduling jobs on machines** Consider the two schedules  $(213, 132, 321)$  and  $(213, 312, 321)$  which are both members of the schema  $(213, *, 321)$ . Since structural crossover works by slicing between the component schedules, any offspring will also belong to this schema. For example, consider the mask  $b = 101$ . We get

$$\begin{aligned} (213, 132, 321) \otimes 101 \oplus 010 \otimes (213, 312, 321) &= (213, 123, 321) \oplus (123, 312, 123) \\ &= (213, 312, 321) \end{aligned}$$

remembering that the schedule 123 is the identity in each component.

### 3.8 A constructive formulation of crossover

The relationship between binary masks and schemata can be used to re-write the definition of the crossover scheme  $\mathcal{C}$  in an alternative form. We will make use of the following lemma:

**Lemma 6** *Given  $u, v, w \in \Omega$  and  $b \in \mathcal{B}$  then  $u \otimes b \oplus \bar{b} \otimes v = w$  if and only if  $u \in \Omega_{\bar{b}} \oplus w$  and  $v \in \Omega_b \oplus w$ .*

**Proof** Suppose  $u \otimes b \oplus \bar{b} \otimes v = w$ . If  $u \in \Omega_{\bar{b}} \oplus w$  then  $v \in \Omega_b \oplus w$  by symmetry. Note that

$$u \in u \oplus \Omega_{\bar{b}} = u \otimes b \oplus \bar{b} \otimes u \oplus \Omega_{\bar{b}} = (w \oplus \bar{b} \otimes v) \oplus \Omega_{\bar{b}} = w \oplus \Omega_{\bar{b}}$$

Conversely, if  $u \in \Omega_{\bar{b}} \oplus w$  then

$$u \otimes b \in (\Omega \otimes \bar{b} \oplus w) \otimes b = \Omega \otimes 0 \oplus (w \otimes b) = \{w \otimes b\}$$

Similarly, if  $v \in \Omega_b \oplus w$  then  $v \otimes \bar{b} = w \otimes \bar{b}$ . Combining these conditions yields

$$u \otimes b \oplus \bar{b} \otimes v = w \otimes b \oplus \bar{b} \otimes w = w$$

□

The previous lemma enables us to re-write the crossover scheme as

$$\begin{aligned} \mathcal{C}(\mathbf{p})_w &= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = w] \\ &= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \in \Omega_{\bar{b}} \oplus w][v \in \Omega_b \oplus w] \\ &= \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{u,v} p_u p_v [u \in \Omega_{\bar{b}} \oplus w][v \in \Omega_b \oplus w] \\ &= \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{u \in \Omega_{\bar{b}} \oplus w} p_u \sum_{v \in \Omega_b \oplus w} p_v \end{aligned}$$

<sup>6</sup>Notice that one-point crossover is not symmetric ( $r(u, v, w) \neq r(v, u, w)$ ) as we always take the left part of the first parent and the right part of the second.

This gives us a “constructive” form of the equation for  $\mathcal{C}(p)$  (in the sense that it indicates how  $w \in \Omega$  is constructed by crossover from elements belonging to pairs of complementary schemata containing  $w$ , (i.e., from  $u \in \Omega_{\bar{b}} \oplus w$  and  $v \in \Omega_b \oplus w$ ) and is a generalization of the schema equation given in (Stephens and Waelbroeck, 1999) for binary strings with one-point crossover.<sup>7</sup>

### 3.9 The form of the mixing matrix

Given mixing matrix  $M$ , define its *skeleton*  $M^\dagger$  to be a  $2^\ell \times 2^\ell$  matrix, indexed over elements of  $\mathcal{B}$ , by

$$M_{a,b}^\dagger = \frac{[a = \bar{b}]}{|Z|} \sum_{z \in Z} M_{z \otimes a, z \otimes b}$$

where  $Z \subset \Omega$  is the set of elements  $z \in \Omega$  for which  $z_i \neq 0$  for all  $i = 0, \dots, \ell - 1$ . Note that the skeleton is indexed by elements  $a, b \in \mathcal{B}$ .

For the example of the fixed-length string representation where  $\Omega$  is isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ , and where crossover is uniform with crossover rate  $c$ ,

$$M^\dagger = \begin{bmatrix} 0 & 0 & 0 & 1/2 - 1/4 c \\ 0 & 0 & 1/4 c & 0 \\ 0 & 1/4 c & 0 & 0 \\ 1/2 - 1/4 c & 0 & 0 & 0 \end{bmatrix}$$

Note that  $M^\dagger$  is always antidiagonal: i.e.,  $M_{a,b}^\dagger = 0$  unless  $a = \bar{b}$ .

Define the matrix  $\Theta$  with rows indexed by  $w \in \Omega$  and columns indexed by  $b \in \mathcal{B}$  by

$$\Theta_{w,b} = [w \in \Omega_{\bar{b}}]$$

For the example where  $\ell = 2$ ,  $c_0 = 3$ ,  $c_1 = 2$ ,

$$\Theta = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

**Lemma 7** *Let mixing matrix  $M$  correspond to a structural crossover operator with distribution  $\chi$ . Then*

$$M_{a,b}^\dagger = [a = \bar{b}] \frac{\chi_a + \chi_b}{2}$$

**Proof** It suffices that for every  $b \in \mathcal{B}$  and for every  $z \in Z$

$$M_{b \otimes z, \bar{b} \otimes z} = \frac{\chi_b + \chi_{\bar{b}}}{2}$$

<sup>7</sup>The effects of selection and mutation are trivial additions since one has simply to compose functions:  $G = C \circ U \circ F$ .

Note that

$$\begin{aligned}
M_{b \otimes z, \bar{b} \otimes z} &= \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} [(b \otimes z) \otimes a \oplus \bar{a} \otimes (\bar{b} \otimes z) = 0] \\
&= \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} ([a = \bar{b}] + [a \neq \bar{b}]) [z \otimes (b \otimes a) \oplus (\bar{a} \otimes \bar{b}) \otimes z = 0] \\
&= \frac{\chi_{\bar{b}} + \chi_b}{2} + \sum_{a \in \mathcal{B}} \frac{\chi_a + \chi_{\bar{a}}}{2} [a \neq \bar{b}] [z \otimes (b \otimes a) \oplus (\bar{a} \otimes \bar{b}) \otimes z = 0]
\end{aligned}$$

The last summation above is zero because if  $a \neq \bar{b}$ , then for some  $i$  either  $(b \otimes a)_i = 1$  or  $(\bar{a} \otimes \bar{b})_i = 1$ . In either case,  $(z \otimes (b \otimes a) \oplus (\bar{a} \otimes \bar{b}) \otimes z)_i = z_i \neq 0$ .  $\square$

**Theorem 8** Suppose crossover scheme  $\mathcal{C}$  commutes with  $\Omega$  and has mixing matrix  $M$ . Then the following are equivalent:

- $\mathcal{C}$  is structural.
- $M = \Theta M^\dagger \Theta^T$

Moreover, in that case a corresponding crossover distribution is given by  $\chi_b = M_{b, \bar{b}}^\dagger$ .

**Proof** First assume that  $\mathcal{C}$  is structural with distribution  $\chi$ . Note that  $\Theta_{w, b} = [w \in \Omega_{\bar{b}}] = [w \otimes b = 0]$ . Using this and lemma 7,

$$\begin{aligned}
(\Theta M^\dagger \Theta^T)_{u, v} &= \sum_{a, b} [u \otimes a = 0] [a = \bar{b}] \frac{\chi_a + \chi_b}{2} [b \otimes v = 0] \\
&= \sum_b \frac{\chi_{\bar{b}} + \chi_b}{2} [u \otimes \bar{b} = 0] [b \otimes v = 0] = \sum_b \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = 0] = M_{u, v}
\end{aligned}$$

Since replacing  $\chi_b$  with  $(\chi_b + \chi_{\bar{b}})/2$  changes neither  $\mathcal{C}$  nor  $M$ , it may be assumed without loss of generality that  $\chi_b = (\chi_b + \chi_{\bar{b}})/2$ , in which case lemma 7 implies that a corresponding crossover distribution is given by  $\chi_b = M_{b, \bar{b}}^\dagger$ .

Conversely, assume that  $M = \Theta M^\dagger \Theta^T$ . This means that

$$M_{u, v} = \sum_{a, b} [u \otimes a = 0] [u = \bar{b}] [b \otimes v = 0] \frac{1}{|Z|} \sum_{z \in Z} M_{z \otimes a, z \otimes b}$$

Let  $\chi_v = |Z|^{-1} \sum_{z \in Z} M_{z \otimes \bar{b}, z \otimes b}$  to simplify the expression above, and note that  $\chi_b = (\chi_b + \chi_{\bar{b}})/2$  since  $M$  is symmetric. It follows that

$$M_{u, v} = \sum_b \chi_b [u \otimes \bar{b} = 0] [b \otimes v = 0] = \sum_b \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = 0]$$

What has been established is that if  $1 = \sum \chi_b$ , then  $\mathcal{C}$  has the same mixing matrix as does a structural crossover scheme having crossover distribution  $\chi$ . Since  $\mathcal{C}$  commutes with  $\Omega$ , these two crossover schemes would therefore coincide. Note that

$$\begin{aligned}
1 &= \frac{1}{2} \sum_w r(u, v, w) + r(v, u, w) = \frac{1}{2} \sum_w r(\ominus w \oplus u, \ominus w \oplus v, 0) + r(\ominus w \oplus v, \ominus w \oplus u, 0) \\
&= \sum_w M_{\ominus w \oplus u, \ominus w \oplus v} = \sum_w \sum_b \frac{\chi_b + \chi_{\bar{b}}}{2} [(\ominus w \oplus u) \otimes b \oplus \bar{b} \otimes (\ominus w \oplus v) = 0] \\
&= \sum_v \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_w [u \otimes b \oplus \bar{b} \otimes v = w] = \sum_b \frac{\chi_b + \chi_{\bar{b}}}{2} = \sum_b \chi_b
\end{aligned}$$

where we have made use of Result 1. □

### 3.10 Example

**Strings over arbitrary alphabets** For the example  $\Omega = \mathcal{Z}_3 \times \mathcal{Z}_2$  under uniform crossover at rate  $c$ , we have already calculated the skeleton  $M^\dagger$  and the corresponding  $\Theta$  matrix. Notice that the antidiagonal of the skeleton matrix contains the appropriate probability distribution over masks for uniform crossover. A calculation verifies that  $M = \Theta M^\dagger \Theta^T$ .

$$\begin{aligned}
 & \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 & 0 & 1/2 - 1/4c \\ 0 & 0 & 1/4c & 0 \\ 0 & 1/4c & 0 & 0 \\ 1/2 - 1/4c & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
 = & \begin{bmatrix} 1 & 1/2 & 1/2 & 1/2 - 1/4c & 1/2 & 1/2 - 1/4c \\ 1/2 & 0 & 1/4c & 0 & 1/4c & 0 \\ 1/2 & 1/4c & 0 & 0 & 0 & 0 \\ 1/2 - 1/4c & 0 & 0 & 0 & 0 & 0 \\ 1/2 & 1/4c & 0 & 0 & 0 & 0 \\ 1/2 - 1/4c & 0 & 0 & 0 & 0 & 0 \end{bmatrix}
 \end{aligned}$$

### 3.11 Structural mutation

Let  $\mu$  be a probability distribution over  $\Omega$ ,

$$\mu_w = \text{the probability of } w \in \Omega$$

**Definition 7** Structural mutation with distribution  $\mu$  applied to element  $v \in \Omega$  corresponds to choosing  $w$  with probability  $\mu_w$  and then producing the result  $v \oplus w$ .

The probability that  $v$  mutates to  $u$  is therefore  $U_{u,v} = \mu_{\Theta v \oplus u}$ . The corresponding mutation scheme  $\mathcal{U}$  is also called structural.

**Theorem 9** A mutation scheme with corresponding matrix  $U$  is structural if and only if it commutes with  $\Omega$ . Moreover, the associated probability distribution is given by  $\mu_w = U_{w,0}$  where  $U$  is the mutation matrix.

**Proof** First assume mutation is structural with distribution  $\mu$ .

$$U_{w \oplus u, w \oplus v} = \mu_{\Theta(w \oplus v) \oplus (w \oplus u)} = \mu_{\Theta v \oplus w \oplus w \oplus u} = \mu_{\Theta v \oplus u} = U_{u,v}$$

By result 2, mutation commutes with  $\Omega$ .

Conversely, assume mutation commutes with  $\Omega$ . Define  $\mu$  by  $\mu_w = U_{w,0}$ . By result 2, the probability that  $v$  mutates to  $u$  is  $U_{u,v} = U_{\Theta v \oplus u, 0} = \mu_{\Theta v \oplus u}$ . □

### 3.12 Independent mutation

The standard mutation of binary strings involves changing bits independently (usually with some probability called the *rate*). In general, structural mutation is called *independent* if for all  $v \in \Omega$  and  $b \in \mathcal{B}$

$$\mu_v = \sum_{u: b \otimes u = 0} \mu_{u \oplus v} \sum_{u: \bar{b} \otimes u = 0} \mu_{u \oplus v}$$

Independent mutation has the nice property that it commutes with structural crossover. That is, it doesn't matter whether you do mutation first and then crossover, or the other way around: the effect on the population is the same.

**Theorem 10** *Let mutation  $\mathcal{U}$  be independent and let crossover  $\mathcal{C}$  be structural. Then  $\mathcal{U} \circ \mathcal{C} = \mathcal{C} \circ \mathcal{U}$ .*

**Proof** By definition, the  $w$  th component of  $\mathcal{C} \circ \mathcal{U}(\mathbf{p})$  is<sup>8</sup>

$$\begin{aligned}
& \sum_{x,y} \sum_u \mu_{\ominus u \oplus x} p_u \sum_v \mu_{\ominus v \oplus y} p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [x \otimes b \oplus \bar{b} \otimes y = w] \\
&= \sum_{u,v} p_u p_v \sum_{x,y} \mu_x \mu_y \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [(u \oplus x) \otimes b \oplus \bar{b} \otimes (v \oplus y) = w] \\
&= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{x,y} \mu_x \mu_y [u \otimes b \oplus x \otimes b \oplus \bar{b} \otimes v \oplus \bar{b} \otimes y = w] \\
&= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{x,y} \mu_x \mu_y [u \otimes b \oplus \bar{b} \otimes v \oplus x \otimes b \oplus \bar{b} \otimes y = w] \\
&= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{x,y} \mu_x \mu_y \sum_z [u \otimes b \oplus \bar{b} \otimes v \oplus z = w] [x \otimes b \oplus \bar{b} \otimes y = z] \\
&= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_z [u \otimes b \oplus \bar{b} \otimes v \oplus z = w] \sum_{x,y} \mu_x \mu_y [x \otimes b \oplus \bar{b} \otimes y = z]
\end{aligned}$$

Note that the inner sum is

$$\begin{aligned}
\sum_{x,y} \mu_x \mu_y [x \otimes b = z \otimes b] [\bar{b} \otimes y = \bar{b} \otimes z] &= \sum_x \mu_x [x \otimes b = z \otimes b] \sum_y \mu_y [\bar{b} \otimes y = \bar{b} \otimes z] \\
&= \sum_{(x \ominus z) \otimes b = 0} \mu_x \sum_{(y \ominus z) \otimes \bar{b} = 0} \mu_y = \sum_{x \otimes b = 0} \mu_{x \oplus z} \sum_{y \otimes \bar{b} = 0} \mu_{y \oplus z}
\end{aligned}$$

Since mutation is independent, the last expression is  $\mu_z$ . It follows that

$$\begin{aligned}
\mathcal{C} \circ \mathcal{U}(\mathbf{p})_w &= \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_z \mu_z [u \otimes b \oplus \bar{b} \otimes v \oplus z = w] \\
&= \sum_z \mu_{\ominus z \oplus w} \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = z] \\
&= \sum_z U_{w,z} \sum_{u,v} p_u p_v \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus \bar{b} \otimes v = z] \\
&= \mathcal{U} \circ \mathcal{C}(\mathbf{p})_w
\end{aligned}$$

□

We would like a general way to implement changing a positional element of an individual according to some rate (so that each position is changed independently). To this end, we let  $\zeta$  be a function  $\zeta : \{0, \dots, \ell - 1\} \times \mathcal{Z} \rightarrow [0, 1]$  with the property that for all  $i \in \{0, \dots, \ell - 1\}$

$$\sum_{h=0}^{|A_i|-1} \zeta(i, h) = 1$$

That is, we get a probability distribution over the elements of  $A_i$  (enumerated in some fashion).

<sup>8</sup>Throughout this proof,  $u, v, w, x, y, z \in \Omega$ .

**Theorem 11** Let  $\zeta$  be a function with the above property. If the probability distribution  $\mu$  over  $\Omega$  is defined by

$$\mu_v = \prod_{i=0}^{\ell-1} \zeta(i, v_i)$$

then  $\mu$  is independent.

The following lemma will be used in the proof of theorem 11.

**Lemma 12** If  $b \in \mathcal{B}$ , then

$$\sum_{v \in \Omega_b} \prod_{i: b_i=1} \zeta(i, v_i) = 1$$

**Proof** The proof is by induction on  $\#b$ , the number of ones in mask  $b$ . The lemma is trivially true for  $\#b = 0$ . Let  $b = a \oplus b'$  with  $\#a = 1$  and  $a \otimes b' = 0$ . Then  $\#b' = \#b - 1$ . Assume that  $\Omega_a = A_j$ . Then

$$\sum_{v \in \Omega_b} \prod_{i: b_i=1} \zeta(i, v_i) = \left( \sum_{h \in A_j} \zeta(j, h) \right) \left( \sum_{v \in \Omega_{b'}} \prod_{i: b'_i=1} \zeta(i, v_i) \right)$$

The left summation is 1 since  $\zeta$  defines a probability distribution over  $A_j$ , and the right summation is 1 by induction.  $\square$

**Proof of theorem 11.** The rightmost factor in the definition of independence is

$$\begin{aligned} \sum_{\bar{b} \otimes u=0} \mu_{u \oplus v} &= \sum_{u \in \Omega_b} \left( \prod_{i: b_i=1} \zeta(i, (u \oplus v)_i) \right) \left( \prod_{i: b_i=0} \zeta(i, (u \oplus v)_i) \right) \\ &= \left( \prod_{i: b_i=0} \zeta(i, v_i) \right) \left( \sum_{u \in \Omega_b} \prod_{i: b_i=1} \zeta(i, (u \oplus v)_i) \right) \\ &= \prod_{i: b_i=0} \zeta(i, v_i) \quad \text{by lemma 12.} \end{aligned}$$

A similar argument can be used to compute the leftmost factor in the definition of independence.

$$\sum_{b \otimes u=0} \mu_{u \oplus v} = \prod_{i: b_i=1} \zeta(i, v_i)$$

Thus,

$$\mu_v = \prod_{i=0}^{\ell-1} \zeta(i, v_i) = \prod_{i: b_i=1} \zeta(i, v_i) \prod_{i: b_i=0} \zeta(i, v_i) = \sum_{b \otimes u=0} \mu_{u \oplus v} \sum_{\bar{b} \otimes u=0} \mu_{u \oplus v}$$

$\square$

**Definition 8** We define mutation by a rate,  $\varepsilon$ , by letting

$$\zeta(i, h) = \begin{cases} 1 - \varepsilon & \text{if } h = 0 \\ \frac{\varepsilon}{|A_i|-1} & \text{if } h \neq 0 \end{cases}$$

**Corollary 13** If  $\mu$  corresponds to a mutation rate, then  $\mu$  is independent.



### 3.13 Examples

**Strings over arbitrary alphabets** Suppose we have some mutation rate  $0 < \mu < 1/2$ .

This gives a probability that any component of a string will mutate. When it does mutate, it changes to one of its other possible values, selected uniformly at random. This kind of mutation by rate is structural and, in fact, independent (see theorem 13).

**Scheduling jobs on machines** Theorem 9 tells us that we can get a structural mutation operator by assigning *any* distribution we like to  $\Omega$ . In the machine scheduling example, we can pick some distribution

$$\begin{aligned} \mu_{(123,321,213)} &= 0.152 & \mu_{(231,321,231)} &= 0.323 \\ \mu_{(312,312,312)} &= 0.124 & \mu_{(312,123,123)} &= 0.222 \\ \mu_{(213,321,231)} &= 0.179 & \mu_v &= 0.000 \text{ otherwise} \end{aligned}$$

Mutation now works by picking an element of  $\Omega$  according to this distribution and applying it on the right. For example, suppose we have schedule  $u = (312, 213, 231)$  and we randomly choose element  $v = (123, 321, 213)$  with which to mutate it. The result is

$$u \oplus v = (312, 213, 231) \oplus (123, 321, 213) = (312, 312, 321)$$

Of course, this kind of mutation operator may not have as natural an interpretation as mutation by a rate.

## 4 Schemata projections

### 4.1 Introduction

This section describes the remarkable properties of structural crossover and mutation with respect to families of competing schemata. Recall that a schema defines certain components as taking on fixed values, while the remaining components are free to vary. Two schemata are in the same family if they specify fixed values for the same set of components. They are competing if they specify different values at these components. Given a set of components, the set of all possible fixed values that can be assigned to them gives us a whole competing family of schemata. Such a set of components may be picked out by a binary mask, and we can view such a mask as being a projection onto a smaller search space (in which we ignore what happens at the other positions). The remarkable result that we will prove is that structural crossover and mutation project naturally onto these families. That is, we can specify the effect of crossover and mutation just on the set of positions under consideration, ignoring what happens at other positions.

The result that operators project naturally onto schemata families applies only to crossover and mutation. It would be nice if it also applied to selection, and this possibility is investigated. The conclusion, however, is that it cannot (except in the trivial case where fitness is a constant for each schema in a schema family). One can, of course, write down an equation which involves selection, as well as crossover and mutation. However, this equation, which is an exact version of Holland's original Schema Theorem, suffers from the same fundamental flaw. In order to compute the average fitness of a schema at a given generation, one needs to know all the details of the entire population at that generation. It is therefore impossible to project onto a schemata family and ignore what happens at the other components. One implication of this is that one

cannot iterate the equation: like the Schema Theorem, it can be applied over one time-step only. There is one exception to this: when the population is at a fixed-point! An example of this is elaborated.

Finally, although one-point crossover, for example, does project naturally onto a schemata family (comprising, say, the first, third and eighth component), one cannot expect it to look like one-point crossover on that family. The situation is different for uniform crossover, which remains uniform crossover on the schemata family. A similar observation applies to mutation by a rate, which is generalised at the end of this section.

#### 4.2 Masks as projections

This section generalizes (Vose and Wright, 2001). Assume  $\Omega$  is structural, crossover is structural, and mutation is structural. Recall that binary mask  $b$  has associated subgroup  $\Omega_b = b \otimes \Omega$  and that  $\mathcal{N}$  is the collection of all such subgroups. The map  $u \mapsto b \otimes u$  is a homomorphism from  $\Omega$  to  $\Omega_b$  since  $b \otimes (u \oplus v) = b \otimes u \oplus b \otimes v$ . The kernel is the normal subgroup  $\Omega_{\bar{b}}$ , and, therefore, the following map from the image  $\Omega_b$  to the quotient group  $\Omega/\Omega_{\bar{b}}$  is an isomorphism (Lang, 1993),

$$w = w \otimes b \mapsto \Omega_{\bar{b}} \oplus w$$

The quotient group

$$\Omega/\Omega_{\bar{b}} = \{\Omega_{\bar{b}} \oplus w : w \in \Omega_b\}$$

being comprised of disjoint schemata, is referred to as the *schema family corresponding to  $b$* , and schema  $\Omega_{\bar{b}} \oplus w$  is referred to as the *schema corresponding to  $w \in \Omega_b$* . For example, in  $\mathcal{Z}_3 \times \mathcal{Z}_2$ , the schema family corresponding to  $b = 10$  is the set of competing schemata  $\{0*, 1*, 2*\}$ . The schema family corresponding to mask 01 is  $\{*0, *1\}$ .

For  $b \in \mathcal{B}$ , define  $\Lambda_b$  as

$$\Lambda_b = \left\{ \mathbf{p} \in \mathbb{R}^{|\Omega_b|} : p_u \geq 0, \sum p_u = 1 \right\}$$

The linear operator  $\Xi_b : \mathbb{R}^{|\Omega|} \rightarrow \mathbb{R}^{|\Omega_b|}$  with matrix

$$(\Xi_b)_{u,v} = [v \otimes b = u]$$

is called the *projection associated with the schema family corresponding to  $b$* ; it has rows indexed by elements of  $\Omega_b$  and columns indexed by  $\Omega$ . Notice that  $\Xi_b(\Lambda) \subseteq \Lambda_b$ . To simplify notation, we will refer simply to  $\Xi$  when the binary mask  $b$  is understood.

For the example of the fixed length string representation where  $\Omega$  is isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ , for  $b = 10$ ,

$$\Xi_{10} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

and for  $b = 01$ ,

$$\Xi_{01} = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}$$

Note that

$$\sum_u (\Xi \mathbf{p})_u = \sum_{u \in \Omega_b} \sum_v [v \otimes b = u] p_v = \sum_v p_v \sum_{u \in \Omega_b} [v \otimes b = u] = \sum_v p_v$$

Hence if  $p \in \Lambda$  is a probability vector, then  $\Xi p \in \Lambda_b$  is a probability vector. As the following computation shows, the  $u$ th component of  $\Xi p$  is simply the proportion of the population  $p$  which is contained in the schema  $\Omega_{\bar{b}} \oplus u$  which corresponds to  $u \in \Omega_b$ ,

$$(\Xi p)_u = \sum_v [v \otimes b = u] p_v = \sum_v [v = v \otimes \bar{b} \oplus u] p_v = \sum_v [v \in \Omega_{\bar{b}} \oplus u] p_v$$

Let  $\mathcal{B}_b = b \otimes \mathcal{B}$ . It is notationally convenient to make  $\Xi_b$  polymorphic by extending it to also represent the linear map  $\Xi_b : \mathbb{R}^{|\mathcal{B}|} \rightarrow \mathbb{R}^{|\mathcal{B}_b|}$  with matrix

$$(\Xi_b)_{a,a'} = [a' \otimes b = a]$$

Here the rows are indexed by elements of  $\mathcal{B}_b$  and columns are indexed by  $\mathcal{B}$ . Again, we will drop the subscript and refer simply to  $\Xi$  when the mask is understood.

For the example of the fixed length string representation where  $\Omega$  is isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ , the set of masks is  $\mathcal{B} = \{00, 01, 10, 11\}$ . For  $b = 10$ ,

$$\Xi_{10} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix},$$

and for  $b = 01$ ,

$$\Xi_{01} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}.$$

Note that

$$\sum_a (\Xi \chi)_a = \sum_{a \in \mathcal{B}_b} \sum_{a'} [a' \otimes b = a] \chi_{a'} = \sum_{a'} \chi_{a'} \sum_{a \in \mathcal{B}_b} [a' \otimes b = a] = \sum_{a'} \chi_{a'}$$

Hence if  $\chi \in \mathbb{R}^{|\mathcal{B}|}$  is a probability vector, then  $\Xi \chi \in \mathbb{R}^{|\mathcal{B}_b|}$  is a probability vector.

### 4.3 The schemata projection theorem

Given that  $\Omega = A_0 \oplus \dots \oplus A_{\ell-1}$  is structural,  $\Omega_b$  is also structural,

$$\Omega_b = A_{k_0} \oplus \dots \oplus A_{k_{\#b-1}}$$

where  $\{k_0, \dots, k_{\#b-1}\} = \{i : b_i = 1\}$ . Moreover,  $\Lambda_b$  is precisely the  $\Lambda$  previously defined as corresponding to the search space, if the search space is chosen to be  $\Omega_b$ . Likewise,  $\mathcal{B}_b$  is precisely the  $\mathcal{B}$  previously defined as corresponding to the search space, if the search space is chosen to be  $\Omega_b$ . Therefore, since  $\Xi \chi$  and  $\Xi \mu$  are probability vectors indexed by  $\mathcal{B}_b$  and  $\Omega_b$  (respectively), they have corresponding structural crossover and mutation schemes  $\mathcal{C}_b$  and  $\mathcal{U}_b$  which represent crossover and mutation on the state space  $\Lambda_b$ .

**Theorem 14** *Let  $\mathcal{U}$  and  $\mathcal{C}$  be structural mutation and crossover schemes, respectively. If  $\mathcal{M} = \mathcal{U} \circ \mathcal{C}$ , then  $\Xi \mathcal{M}(p) = \mathcal{U}_b \circ \mathcal{C}_b(\Xi p)$*

**Proof** See Theorem 1 of (Wright et al., 2003). □

#### Corollary 15 (Schemata Projection Theorem)

$$\mathcal{M} = \mathcal{U} \circ \mathcal{C} \implies \Xi \mathcal{M} = \mathcal{M}_b \circ \Xi \text{ where } \mathcal{M}_b = \mathcal{U}_b \circ \mathcal{C}_b$$

$$\mathcal{M} = \mathcal{C} \circ \mathcal{U} \implies \Xi \mathcal{M} = \mathcal{M}_b \circ \Xi \text{ where } \mathcal{M}_b = \mathcal{C}_b \circ \mathcal{U}_b$$

*In particular,  $\Xi \mathcal{U} = \mathcal{U}_b \circ \Xi$  and  $\Xi \mathcal{C} = \mathcal{C}_b \circ \Xi$ .*

**Proof** See Corollary 1 in (Wright et al., 2003). □

Corollary 15 speaks to schemata through the isomorphism  $\Omega_b \cong \Omega/\Omega_{\bar{b}}$  given by  $u \mapsto \Omega_{\bar{b}} \oplus u$ . Therefore,  $\mathcal{M}_b$  represents mixing (i.e., crossover and mutation) on a search space of schemata (i.e., the schema family  $\Omega/\Omega_{\bar{b}}$ ). Since crossover and mutation are structural, theorems 1 and 9 imply that  $\mathcal{C}_b$ ,  $\mathcal{U}_b$ , and  $\mathcal{M}_b$  commute with  $\Omega_b$  (i.e., they commute with  $\Omega/\Omega_{\bar{b}}$ ) and theorem 3 implies that  $\mathcal{C}_b$  respects the  $\mathcal{N}$ -schemata of  $\Omega_b$  (i.e., of  $\Omega/\Omega_{\bar{b}}$ ).

A consequence of corollary 15 is that, independent of the order of crossover and mutation, *the following commutative diagram holds, in parallel, for every choice of schema family, simultaneously*

$$\begin{array}{ccc} p & \longrightarrow & \mathcal{M}(p) \\ \Xi \downarrow & & \downarrow \Xi \\ \Xi p & \longrightarrow & \mathcal{M}_b(\Xi p) \end{array}$$

#### 4.4 Historical note

Early studies of genetic algorithms attributed their success to something called “implicit” or “intrinsic” parallelism. This idea came from an interpretation of the Schema Theorem in which selection was seen to be acting simultaneously on many schemata (selecting those of above average fitness). This view is, of course, now known to be incorrect (see, for example, (Wright et al., 2003) for detailed analysis). As will be shown in section 4.6, proportional selection does not project onto schemata families. However, the Schemata Projection Theorem tells us that structural crossover and mutation do have this nice property. This has led Vose (Vose, 1999) to redefine the term “implicit parallelism” to apply to this correct result.<sup>9</sup> The Schemata Projection Theorem does not, however, imply that GAs enjoy any kind of “processing leverage”.

#### 4.5 Examples

**Strings over arbitrary alphabets** Let us consider the schemata projection of mutation on the example  $\Omega = \mathcal{Z}_3 \times \mathcal{Z}_2$ . Firstly, let us define our mutation operator by the probability distribution:

$$\mu_v = \begin{cases} 0.9 & \text{if } v = 00 \\ 0.02 & \text{otherwise} \end{cases}$$

That is, there is a probability of 0.9 that no mutation will take place. Otherwise, we pick an element  $v \in \Omega \setminus \{00\}$  at random (uniformly) and apply it to our current individual. Now suppose that we are interested in what happens in the first component. That is, we are concerned with the effect of mutation on the family of schemata  $0*, 1*, 2*$ . One way to calculate this would be to work out the effect of mutation on the whole population and then sum up the results for each schema in the family. The schemata projection theorem tells us that we don’t need to do this. Instead, we can find a mutation operator that acts on the family of schemata itself, and has the exact equivalent effect.

For a concrete example, consider the population vector

$$p = (p_{00}, p_{01}, p_{10}, p_{11}, p_{20}, p_{21}) = (0.1, 0.2, 0.1, 0.2, 0.25, 0.15)$$

---

<sup>9</sup>... in the binary case. This paper establishes the result more generally.

Our family of schemata corresponds to the mask  $b = 10$ . We have already seen that this gives us a matrix

$$\Xi_{10} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

Multiplying  $p$  by this matrix gives us the distribution of the population over the family of schemata:  $\Xi_{10}p = (p_{0*}, p_{1*}, p_{2*}) = (0.3, 0.3, 0.4)$ . We now have to define a mutation operator for this reduced search space. This is given by  $\Xi_{10}\mu = (0.92, 0.04, 0.04)$ . So our mutation operator acting on our family of schemata consists of picking an element of  $\{0*, 1*, 2*\}$  according to the above probability distribution and applying it to the element to be mutated. Notice that in this quotient group the element  $0*$  is the identity. Constructing the mutation operator that acts on  $\Lambda_b$  from this distribution gives us

$$\mathcal{U}_{10}(p) = \begin{bmatrix} 0.92 & 0.04 & 0.04 \\ 0.04 & 0.92 & 0.04 \\ 0.04 & 0.04 & 0.92 \end{bmatrix} p$$

So in our example, we calculate the effect of mutation on the family of schemata as being

$$\begin{bmatrix} 0.92 & 0.04 & 0.04 \\ 0.04 & 0.92 & 0.04 \\ 0.04 & 0.04 & 0.92 \end{bmatrix} \begin{bmatrix} 0.3 \\ 0.3 \\ 0.4 \end{bmatrix} = \begin{bmatrix} 0.304 \\ 0.304 \\ 0.392 \end{bmatrix}$$

Notice that to make this calculation we did not need to know the details of the population  $p$ . We only needed to know how many elements were in each schema (given by  $\Xi p$ ). We can check this result by working out the effect of mutation on the whole population and then summing over the schemata. The schemata projection theorem tells us that we will get exactly the same result.

**Scheduling jobs on machines** Now we will consider the schemata projection of crossover on the job scheduling problem  $\Omega = S_3 \times S_3 \times S_3$ . We will use one-point crossover (at rate  $1/2$ ), given by the following distribution:

$$\chi_{000} = 1/4, \chi_{100} = 1/4, \chi_{110} = 1/4, \chi_{111} = 1/4$$

with  $\chi_b = 0$  for other masks. Suppose we are interested in the schemata family  $\{(*, 123, *), (*, 132, *), (*, 213, *), (*, 231, *), (*, 312, *), (*, 321, *)\}$ . This corresponds to the mask  $b = 010$ . The operator  $\Xi_{010}$  has the effect of summing over these schemata. We could calculate the effect of crossover on this family by working it out for the entire population  $p$  and then summing. However, the schemata projection theorem tells us that we don't need to do this. We will define a new crossover operator on the reduced search space, and then this will act on the reduced vector  $\Xi_{010}p$ . The new crossover operator is found by calculating  $\Xi_{010}\chi$  and then using this as a crossover distribution in the reduced search space. Remember that when we apply  $\Xi_{010}$  to  $\chi$  we are using its alternative (polymorphic) definition.

We find, then, that the new crossover distribution is

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} 1/4 \\ 0 \\ 0 \\ 0 \\ 1/4 \\ 0 \\ 1/4 \\ 1/4 \end{bmatrix} = \begin{bmatrix} 1/2 \\ 1/2 \end{bmatrix}$$

That is, we have two possible masks in our reduced set of masks, namely  $*0*$  and  $*1*$  and they each have a probability of  $1/2$  of being selected. The operator  $\mathcal{C}_{010}$  is constructed from this crossover operator. We can now calculate the effect of crossover on this family of schemata, without having to keep track of all the detailed structure of the population.

#### 4.6 Fitness-based selection

It would be especially useful if, in the commutative diagram above,  $\mathcal{M}$  could be generalized to  $\mathcal{G}$  so the effects of selection could be incorporated. For proportional selection at least, Vose has pointed out the difficulties involved and concluded that such commutativity is in general not possible (Vose, 1999). In an attempt to *force* commutativity, a selection scheme  $\mathcal{F}_b$  might be defined on the quotient by

$$\mathcal{F}_b(\Xi x) = \Xi \mathcal{F}(x)$$

The problem here is that  $\mathcal{F}_b$  is not well defined; the right hand side might depend on the particular  $x$  involved even though the left hand side does not (i.e., even if  $\Xi x$  does not). In an attempt to ignore this complication, one might define a “fitness vector”  $f_b$  (over  $\Omega_b$ ) for which

$$\mathcal{F}_b(\Xi x) = \frac{\text{diag}(f_b) \Xi x}{f_b^T \Xi x}$$

Since the complication *cannot* be ignored, the vector  $f_b$  *must depend on*  $x$ . If  $f_b$  is defined as

$$f_b = \text{diag}(\Xi x)^{-1} \Xi \text{diag}(f) x$$

then

$$f_b^T \Xi x = \sum_i (\Xi x)_i^{-1} (\Xi \text{diag}(f) x)_i (\Xi x)_i = \sum_i (\Xi \text{diag}(f) x)_i = \sum_j (\text{diag}(f) x)_j = f^T x$$

Therefore, by way of notational sleight of hand,

$$\mathcal{F}_b(\Xi x) = \frac{\text{diag}(f_b) \Xi x}{f_b^T \Xi x} = \frac{\text{diag}(\text{diag}(\Xi x)^{-1} \Xi \text{diag}(f) x) \Xi x}{f_b^T \Xi x} = \frac{\Xi \text{diag}(f) x}{f_b^T \Xi x} = \Xi \mathcal{F}(x)$$

Of course, this definition for  $f_b$  is precisely the one given in the “schema theorem” (Holland, 1975). Using this definition, one could define  $\mathcal{G}_b = \mathcal{M}_b \circ \mathcal{F}_b$  and conclude

$$\Xi \mathcal{G}(x) = \Xi \mathcal{M} \circ \mathcal{F}(x) = \mathcal{M}_b \circ \Xi \circ \mathcal{F}(x) = \mathcal{M}_b \circ \mathcal{F}_b(\Xi x) = \mathcal{G}_b(\Xi x)$$

thereby “extending” the schemata projection theorem from  $\mathcal{M}$  to  $\mathcal{G}$ . However, it should be noted that because  $f_b$  depends on  $x$ , the “extension” speaks only to what happens

over a single time step (like Holland's result) and the information provided is insufficient to characterize the next generation (even in  $\Lambda_b$ ). The "fitness" of schemata cannot be well-defined (it is not an attribute of schemata  $\Xi x$ , but is determined instead by  $x$ ). In particular, it cannot be used to map out population trajectories.

#### 4.7 Example

**Strings over arbitrary alphabets** Consider the family of schemata  $0^*$ ,  $1^*$ ,  $2^*$  (corresponding to mask  $b = 10$ ) on the search space  $\Omega = \mathcal{Z}_3 \times \mathcal{Z}_2$ . Let the fitness vector be  $f = (f_{00}, f_{01}, f_{10}, f_{11}, f_{20}, f_{21})$ . Then  $f_b$  can be calculated as:

$$f_b = \begin{bmatrix} \frac{f_{00}x_{00}+f_{01}x_{01}}{x_{00}+x_{01}} \\ \frac{f_{10}x_{10}+f_{11}x_{11}}{x_{10}+x_{11}} \\ \frac{f_{20}x_{20}+f_{21}x_{21}}{x_{20}+x_{21}} \end{bmatrix}$$

We can verify that  $\mathcal{F}_b(\Xi x) = \Xi \mathcal{F}(x)$ .

$$\mathcal{F}_b(\Xi x) = \begin{bmatrix} \frac{f_{00}x_{00}+f_{01}x_{01}}{x_{00}+x_{01}} & 0 & 0 \\ 0 & \frac{f_{10}x_{10}+f_{11}x_{11}}{x_{10}+x_{11}} & 0 \\ 0 & 0 & \frac{f_{20}x_{20}+f_{21}x_{21}}{x_{20}+x_{21}} \end{bmatrix} \begin{bmatrix} x_{00} + x_{01} \\ x_{10} + x_{11} \\ x_{20} + x_{21} \end{bmatrix} = \Xi \mathcal{F}(x)$$

#### 4.8 Schemata projections at fixed-points

As indicated in the previous section, a population-independent extension from  $\mathcal{M}$  to  $\mathcal{G}$  depends upon well-defined fitness  $f_b$  for schemata  $\Omega/\Omega_b$ . Put another way, if the fitness of schemata did not change from generation to generation—which in general is untrue—then  $f_b$  would be independent of the population and any population  $p$  on an evolutionary trajectory could be used to define  $f_b = \text{diag}(\Xi x)^{-1} \Xi \text{diag}(f) x$ . Suppose that the population vector  $p$  is a fixed-point of the evolution equations. Because the only point on an evolutionary trajectory beginning from fixed point  $p$  is  $p$  itself, the fitness of schemata do not change. Therefore, schemata projection could be applied at a fixed point  $p$  to conclude

$$\Xi p = \mathcal{M}_b \left( \frac{\text{diag}(f_b) \Xi p}{f_b^T \Xi p} \right)$$

This observation should be tempered, however, by the realization that the equality above cannot be used to solve for fixed points  $\Xi p$  in the quotient  $\Omega_b$  unless  $f_b$ —which depends on  $p$ —is a known function of  $\Xi p$ , that is, unless  $\Xi \text{diag}(f) p$  is a known function of  $\Xi p$ .

The observations above (schemata projection at fixed points) are illustrated with the following example. Consider the search space of length  $\ell$  binary strings under bitwise addition modulo 2,  $\Omega = \mathcal{Z}_2 \times \dots \times \mathcal{Z}_2$ . Note that in this case  $\mathcal{B} = \Omega$ . Consider schema families corresponding to binary masks  $b$  with  $\#b = 1$ . These are the "order 1" schemata. For example, if  $\ell = 4$  and  $b = 0001$ , then the schema family associated with  $b$  is, using traditional schema notation (Goldberg, 1989),  $\Omega_{0001} \cong \Omega/\Omega_{1110} = \{** * 0, *** 1\}$ . More generally, the schemata families are all isomorphic to  $\mathcal{Z}_2$ , and for any  $b$  with  $\#b = 1$ ,  $\Lambda_b = \{\langle p, 1-p \rangle : 0 \leq p \leq 1\}$ . Suppose population  $p$  is a fixed-point of  $\mathcal{G}$ , and denote  $\Xi p$  by  $q \in \Lambda_b$ . At this level of generality,  $\text{diag}(\Xi p)^{-1} \Xi \text{diag}(f) p$  is not a known function of  $q$ , hence let  $f_b = \langle f_0(p), f_1(p) \rangle$ , where the notation indicates a dependence of  $f_b$  on  $p$ . That is,  $f_0(p)$  is the fitness of the schema having a 0 in the

position indicated by mask  $b$ , and  $f_1(\mathbf{p})$  is the fitness of the schema with a 1 in that position, relating to population  $p$ . Since  $\Omega_b$  is a one-bit space, it doesn't matter what form structural crossover takes; the crossover operator  $C_b$  must have mixing matrix

$$\begin{bmatrix} 1 & 1/2 \\ 1/2 & 0 \end{bmatrix}$$

Assume a mutation rate of  $\varepsilon$ . By theorem 16, the mutation operator  $\mathcal{U}_b$  has matrix

$$\begin{bmatrix} 1 - \varepsilon & \varepsilon \\ \varepsilon & 1 - \varepsilon \end{bmatrix}$$

The mixing matrix for  $\mathcal{M}_b$  is therefore

$$\begin{bmatrix} 1 - \varepsilon & 1/2 \\ 1/2 & \varepsilon \end{bmatrix}$$

and the fixed point equation

$$q = \mathcal{M}_b \left( \frac{\text{diag}(f_b) q}{f_b^T q} \right)$$

has right hand side

$$\left\langle \frac{f_0(\mathbf{p})q_0(1 - \varepsilon) + f_1(\mathbf{p})q_1\varepsilon}{f_0(\mathbf{p})q_0 + f_1(\mathbf{p})q_1}, \frac{f_0(\mathbf{p})q_0\varepsilon + f_1(\mathbf{p})q_1(1 - \varepsilon)}{f_0(\mathbf{p})q_0 + f_1(\mathbf{p})q_1} \right\rangle$$

The fixed point equation is invariant under scaling  $f_b$ . Therefore, assume without loss of generality that  $f_0(\mathbf{p})q_0 + f_1(\mathbf{p})q_1 = 1$ . In that case  $f_1(\mathbf{p})q_1 = 1 - f_0(\mathbf{p})q_0$  and the first component of the fixed point equation becomes

$$q_0 = f_0(\mathbf{p})q_0(1 - \varepsilon) + (1 - f_0(\mathbf{p})q_0)\varepsilon = f_0(\mathbf{p})q_0(1 - 2\varepsilon) + \varepsilon$$

which is linear in  $q_0$ . Hence

$$q_0 = \frac{\varepsilon}{1 - f_0(\mathbf{p})(1 - 2\varepsilon)}$$

Similarly,

$$q_1 = \frac{\varepsilon}{1 - f_1(\mathbf{p})(1 - 2\varepsilon)}$$

Rearranging these equations and combining them yields:

$$\frac{f_0(\mathbf{p})}{f_1(\mathbf{p})} = \frac{q_1}{q_0} \left( \frac{q_0 - \varepsilon}{q_1 - \varepsilon} \right)$$

This ratio is independent of the fitness scaling factor introduced earlier and therefore gives the ratio of the fitnesses of the two competing schemata at the fixed-point.

Let us take a concrete example. Consider the search space  $\Omega = \{00, 01, 10, 11\}$  under uniform crossover (with rate 1.0), proportional selection, and bitwise mutation with mutation rate  $\varepsilon = 0.006$ . Let the fitness vector be  $f = \langle 4.00, 4.10, 0.10, 4.11 \rangle$ . The stable fixed-point of the system is  $p = \langle 0.032, 0.737, 0.005, 0.226 \rangle$ . The proportion of the population in the schema \*0 is thus  $q_0 = (\Xi p)_0 = 0.032 + 0.005 = 0.037$  and the



proportion of the population in the schema \* 1 is  $q_1 = (\Xi p)_1 = 0.737 + 0.226 = 0.963$ , where

$$\Xi = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix}$$

Applying our formula, the ratio of the fitnesses of these two schemata should be:  $\frac{0.963}{0.037} \left( \frac{0.037-0.006}{0.963-0.006} \right) = 0.843$ . The quotient fitness is given by  $\text{diag}(\Xi p)^{-1} \Xi \text{diag}(f) p$ , thus the fitness of \* 0 is  $f_0(p) = (4 * 0.032 + 0.1 * 0.05)/0.037 = 3.460$  and the fitness of \* 1 is  $f_1(p) = (4.1 * 0.737 + 4.11 * 0.226)/0.963 = 4.102$ . The ratio of these is 0.843 as predicted. Incidentally, this example provides a counter-example to the *building block hypothesis*: The schema \* 1 has schemata utility of  $f_1(p) = 4.102$  which is a significant fitness advantage compared with the schemata utility of \* 0 given by  $f_0(p) = 3.46$ . Yet the whole system is in a stable equilibrium!

#### 4.9 Uniform crossover and mutation by a rate

It is of interest to further clarify the relationship between the mutation and crossover distributions  $\mu$  and  $\chi$  and their counterparts  $\Xi\mu$  and  $\Xi\chi$ . It is *not* generally true that if the crossover distribution  $\chi$  corresponds to 1-point or 2-point crossover then so does the crossover distribution  $\Xi\chi$ . When the crossover distribution is very sparse (as for 1-point and 2-point crossover),  $\Xi\chi$  might be computed for string lengths approaching a million by summing over schemata. In other cases where that is infeasible, the components of  $\Xi\chi$  must be determined analytically. The situation for mutation by a rate and uniform crossover is considerably nicer, however.

The mutation distribution  $\mu$  corresponding to mutation rate  $\varepsilon$  is

$$\mu_v = (1 - \varepsilon)^{\ell - \#v} \prod_{i: v_i \neq 0} \varepsilon / (|A_i| - 1)$$

where we extend the definition of  $\#$  to count the number of components of an element of  $\Omega$  which are not the identity in that component subgroup.

The crossover distribution  $\chi$  corresponding to uniform crossover with rate  $c$  is

$$\chi_b = \frac{c[b \neq 0]}{|\mathcal{B}|} + (1 - c + \frac{c}{|\mathcal{B}|})[b = 0]$$

**Theorem 16** *If the mutation distribution  $\mu$  corresponds to a mutation rate, then so does the mutation distribution  $\Xi\mu$ . If the crossover distribution  $\chi$  is uniform then so is the crossover distribution  $\Xi\chi$ . Moreover, the mutation and crossover rates are unchanged.*

**Proof** Consider first mutation with rate  $\varepsilon$ .

$$\begin{aligned} (\Xi\mu)_u &= \sum_{v \in \Omega_{\bar{v}}} (1 - \varepsilon)^{\ell - \#(v \oplus u)} \prod_{k: (v \oplus u)_k \neq 0} \varepsilon / (|A_k| - 1) \\ &= (1 - \varepsilon)^{\#b - \#u} \left( \prod_{k: u_k \neq 0} \varepsilon / (|A_k| - 1) \right) \sum_{v \in \Omega_{\bar{v}}} (1 - \varepsilon)^{\#\bar{b} - \#v} \prod_{k: v_k \neq 0} \varepsilon / (|A_k| - 1) \end{aligned}$$

Note that the inner sum is

$$\begin{aligned}
 & \prod_{k:\bar{b}_k \neq 0} \sum_{v_k \in A_k} (1 - \varepsilon)^{[v_k=0]} \left( \frac{\varepsilon}{|A_k| - 1} \right)^{[v_k \neq 0]} \\
 = & \prod_{k:\bar{b}_k \neq 0} \left( (|A_k| - 1)(1 - \varepsilon)^{1-1} \left( \frac{\varepsilon}{|A_k| - 1} \right)^1 + (1 - \varepsilon)^{1-0} \left( \frac{\varepsilon}{|A_k| - 1} \right)^0 \right) \\
 = & 1
 \end{aligned}$$

Next consider uniform crossover with crossover rate  $c$ . Let  $a \in \mathcal{B}_b$ .

$$\begin{aligned}
 (\Xi_b \chi)_a &= \sum_{a' \in \mathcal{B}} [a = b \otimes a'] \left( \frac{c[a' \neq 0]}{|\mathcal{B}|} + (1 - c + \frac{c}{|\mathcal{B}|})[a' = 0] \right) \\
 &= \sum_{a' \in \mathcal{B}} [a = b \otimes a'] \left( \frac{c}{|\mathcal{B}|} + (1 - c)[a' = 0] \right) \\
 &= \frac{c}{|\mathcal{B}|} \sum_{a' \in \mathcal{B}} [a = b \otimes a] + (1 - c) \sum_{a' \in \mathcal{B}} [a = b \otimes a'] [a' = 0] \\
 &= \frac{c|\mathcal{B}_b|}{|\mathcal{B}|} + (1 - c)[a = 0] \\
 &= \frac{c}{|\mathcal{B}_b|} + (1 - c)[a = 0] \\
 &= \frac{c[a \neq 0]}{|\mathcal{B}_b|} + (1 - c + \frac{c}{|\mathcal{B}_b|})[a = 0]
 \end{aligned}$$

□

Moreover, mutation by a rate is independent and therefore commutes with any structural crossover operator, as shown in corollary 13.

#### 4.10 Examples

**Strings over arbitrary alphabets** Uniform crossover, defined on strings over arbitrary alphabets, works exactly as expected. For each component the offspring inherits the corresponding value from one of the two parents, chosen at random. Mutation by a rate is also straightforward. For each component there is a probability of  $\varepsilon$  of it mutating. If it does mutate, it changes to another possible value, chosen uniformly at random.

**Scheduling jobs on machines** In this example, uniform crossover and mutation by a rate work in exactly the same way as in the previous example. Technically, mutation works by choosing an element of the component subgroup at random and applying it to the current element. But this is just the same as replacing the current element of the subgroup with another, chosen uniformly at random.

## 5 The Fourier transform and Abelian groups

### 5.1 Introduction

It has been known for some time that the Walsh basis is particularly convenient for analysing crossover and mutation, when the search space is fixed-length binary strings (see, for example, (Vose and Wright, 1998)). One property that is especially important is

that each of the permutation matrices associated with crossover become diagonalised when expressed in the Walsh basis. It is natural to ask whether or not a similar basis exists for other search spaces, and this section answers that question. The properties of the Walsh basis turn out to be intimately connected with the group structure defined on the set of binary strings (under component-wise exclusive-or). Generalising, we find that for Abelian groups we can construct a Fourier transform, which generalises the Walsh transform. In this new basis, the permutation matrices associated with crossover are again diagonalised. However, it turns out that for non-Abelian groups it is not possible to construct such a basis. So, for example, if the search space is identified with the group of all permutations of an underlying set (as with the travelling salesman problem) then there can be no equivalent to the Walsh basis, as such groups are not commutative. The class of Abelian search spaces is characterised by being representable as strings over arbitrary alphabets.

## 5.2 Abelian groups

If the search space group  $(\Omega, \oplus)$  is Abelian, then it is structural and is isomorphic to a direct product of cyclic groups (Lang, 1993),  $\Omega \cong \mathcal{Z}_{c_0} \times \dots \times \mathcal{Z}_{c_{\ell-1}}$ . Here  $\mathcal{Z}_{c_i}$  is the group  $\{0, 1, \dots, c_i - 1\}$  of integers under addition modulo  $c_i$ . Let the cardinality of  $\Omega$  be  $n$ . That is,  $n = \prod_{i=0}^{\ell-1} c_i$ .

Let  $\mathcal{S}$  be the unit circle in the complex plane  $\mathcal{S} = \{z \in \mathbf{C} : zz^* = 1\}$  where  $z^*$  indicates the complex conjugate of  $z$ . For each  $i = 0, \dots, \ell - 1$ , define  $\psi_i : \mathcal{Z} \rightarrow \mathcal{S}$  by

$$\psi_i(x) = \exp\left(\frac{2x\pi\sqrt{-1}}{c_i}\right)$$

It is straightforward to show, for any  $x, y \in \mathcal{Z}$ , that  $\psi_i(x) = \psi_i(x \bmod c_i)$  and  $\psi_i(x+y) = \psi_i(x)\psi_i(y)$ . Define  $\alpha : \Omega \rightarrow \mathcal{S}$  by

$$\alpha(v) = \prod_{i=0}^{\ell-1} \psi_i(v_i)$$

**Lemma 17** *The map  $\alpha : \Omega \rightarrow \mathcal{S}$  is a homomorphism from the group  $(\Omega, \oplus)$  to the group  $(\mathcal{S}, \cdot)$ , where  $\cdot$  indicates complex multiplication. In particular,  $\alpha(u \oplus v) = \alpha(u)\alpha(v)$  and  $\alpha(\ominus v) = \alpha(v)^*$ .*

**Proof**

$$\alpha(u \oplus v) = \prod_{i=0}^{\ell-1} \psi_i((u \oplus v)_i) = \prod_{i=0}^{\ell-1} \psi_i(u_i + v_i) = \prod_{i=0}^{\ell-1} \psi_i(u_i)\psi_i(v_i) = \alpha(u)\alpha(v)$$

Therefore,  $\alpha(v)\alpha(v)^* = 1 = \alpha(0) = \alpha(v \ominus v) = \alpha(v)\alpha(\ominus v) \implies \alpha(v)^* = \alpha(\ominus v)$ .  $\square$

Extend the multiplication operators on the  $\mathcal{Z}_{c_i}$  to a component-wise multiplication operator  $\otimes$  on  $\Omega$ :

$$u \otimes v = \langle u_0 v_0 \bmod c_0, \dots, u_{\ell-1} v_{\ell-1} \bmod c_{\ell-1} \rangle$$

It follows that, for all  $u, v, w \in \Omega$

$$\begin{aligned} (u \oplus v) \otimes w &= (u \otimes w) \oplus (v \otimes w) \\ \ominus(u \otimes w) &= (\ominus u) \otimes w = u \otimes (\ominus w) \end{aligned}$$

**Lemma 18** For any  $w \in \Omega$ ,

$$\sum_{v \in \Omega} \alpha(w \otimes v) = n[w = 0]$$

(recall that  $n = |\Omega|$ ).

**Proof** The case  $w = 0$  is trivial, assume therefore that  $w \neq 0$ . The sum is

$$\sum_{v \in \Omega} \alpha(w \otimes (v_0 \oplus \dots \oplus v_{\ell-1})) = \prod_{i=0}^{\ell-1} \sum_{v_i=0}^{c_i-1} \alpha(w \otimes v_i)$$

where, by abuse of notation,  $v_i$  also represents the element of  $\Omega$  that has the same value as  $v$  at position  $i$ , and is the identity on all other positions. Since  $w \neq 0$ , let  $i$  be such that  $w_i \neq 0$  and consider the following factor in the product above

$$\sum_{v_i=0}^{c_i-1} \alpha(w \otimes v_i) = \sum_{v_i=0}^{c_i-1} \psi_i(w_i v_i) = \sum_{v_i=0}^{c_i-1} \exp\left(\frac{2\pi w_i \sqrt{-1}}{c_i} v_i\right) = \frac{\exp\left(\frac{2\pi w_i \sqrt{-1}}{c_i} c_i\right) - 1}{\exp\left(\frac{2\pi w_i \sqrt{-1}}{c_i}\right) - 1}$$

by summing the geometric series. Note that the numerator is zero. □

### 5.3 Example

**Strings over arbitrary alphabets** The group  $\Omega = \mathbb{Z}_3 \times \mathbb{Z}_2$  is Abelian. We therefore define  $\alpha : \Omega \rightarrow \mathcal{S}$  by

$$\begin{aligned} \alpha(v) &= \exp\left(\frac{2v_0\pi\sqrt{-1}}{3}\right) \exp\left(\frac{2v_1\pi\sqrt{-1}}{2}\right) \\ &= \exp\left(\frac{2v_0 + 3v_1}{3}\pi\sqrt{-1}\right) \end{aligned}$$

For example  $\alpha(21) = \exp(7\pi\sqrt{-1}/3)$ . Lemma 18 can be illustrated by the example:

$$\begin{aligned} &\alpha(21 \otimes 00) + \alpha(21 \otimes 01) + \alpha(21 \otimes 10) + \alpha(21 \otimes 11) + \alpha(21 \otimes 20) + \alpha(21 \otimes 21) \\ &= \alpha(00) + \alpha(01) + \alpha(20) + \alpha(21) + \alpha(10) + \alpha(11) \\ &= \exp(0) + \exp(\pi\sqrt{-1}) + \exp\left(\frac{4\pi\sqrt{-1}}{3}\right) \\ &\quad + \exp\left(\frac{7\pi\sqrt{-1}}{3}\right) + \exp\left(\frac{2\pi\sqrt{-1}}{3}\right) + \exp\left(\frac{5\pi\sqrt{-1}}{3}\right) \end{aligned}$$

which is the sum of all the distinct sixth roots of unity and so equals zero.

### 5.4 The Fourier transform

**Definition 9** Given an Abelian search space group  $\Omega \cong \mathbb{Z}_{c_0} \times \dots \times \mathbb{Z}_{c_{\ell-1}}$ , define the Fourier transform  $W$  of the group to be the matrix

$$W_{u,v} = \frac{\alpha(u \otimes v)}{\sqrt{n}}$$

where  $W$  is indexed by  $u, v \in \Omega$ , and  $n = |\Omega|$ .

For example, where the search space is isomorphic to  $\mathcal{Z}_3 \times \mathcal{Z}_2$ ,

$$W = \frac{\sqrt{6}}{6} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & 1 & -R^* & -R^* & -R & -R \\ 1 & -1 & -R^* & R^* & -R & R \\ 1 & 1 & -R & -R & -R^* & -R^* \\ 1 & -1 & -R & R & -R^* & R^* \end{bmatrix}$$

where  $R = \exp(\pi\sqrt{1}/3) = \frac{1}{2}(1 + \sqrt{-3})$ .

**Lemma 19** *The Fourier transform is unitary.*

**Proof** From the definition of  $\otimes$ , the matrix  $W$  is symmetric. It therefore suffices to show that  $W^{-1} = W^*$ . The  $u, v$  th component of  $WW^*$  is

$$\begin{aligned} \frac{1}{n} \sum_{w=0}^{n-1} \alpha(u \otimes w) \alpha(\ominus v \otimes w) &= \frac{1}{n} \sum_{w=0}^{n-1} \alpha(u \otimes w \ominus v \otimes w) = \frac{1}{n} \sum_{w=0}^{n-1} \alpha((u \ominus v) \otimes w) \\ &= [u \ominus v = 0] \end{aligned}$$

(from Lemma 18). Therefore  $WW^*$  is the identity matrix.  $\square$

**Definition 10** *The Fourier transform  $\widehat{A}$  of a  $n \times n$  matrix  $A$  is  $\widehat{A} = WA^*W^*$ . The Fourier transform  $\widehat{x}$  of a (column) vector  $x$  is  $\widehat{x} = Wx^*$ . The Fourier transform  $\widehat{y}$  of a row vector  $y$  is  $\widehat{y} = y^*W^*$ .*

The following theorem follows easily from these definitions and lemma 19.

**Theorem 20** *Let  $x$  be a column vector,  $y$  a row vector, and  $A$  a square matrix. If  $X$  and  $Y$  are any of these, and if superscript  $H$  denotes conjugate transpose, then  $\widehat{\widehat{X}} = X$ ,  $\widehat{\widehat{X+Y}} = \widehat{X} + \widehat{Y}$ ,  $\widehat{\widehat{XY}} = \widehat{X}\widehat{Y}$ ,  $\widehat{\widehat{X}^H} = \widehat{X}^H$ ,  $\widehat{\widehat{X}^{-1}} = \widehat{X}^{-1}$ ,  $\text{spec}(\widehat{X}) = \text{spec}(\widehat{X})^*$  whenever operations are defined.*

**Theorem 21** *If  $\Omega$  is Abelian, then  $\widehat{\sigma}_w$  is diagonal for all  $w \in \Omega$ .*

**Proof** The  $u, v$  th entry of  $\widehat{\sigma}_w$  is

$$\begin{aligned} \frac{1}{n} \sum_{x,y} \alpha(u \otimes x) [x \oplus w = y] \alpha(\ominus y \otimes v) &= \frac{1}{n} \sum_x \alpha(u \otimes x \ominus (x \oplus w) \otimes v) \\ &= \frac{1}{n} \sum_x \alpha((u \ominus v) \otimes x \ominus w \otimes v) = \alpha(\ominus w \otimes v) [u \ominus v = 0] \end{aligned}$$

where  $x, y \in \Omega$ .  $\square$

**Theorem 22** *Given search space group  $\Omega$ , suppose there exists a matrix  $W$  such that  $W\sigma_w W^{-1}$  is diagonal for all  $w \in \Omega$ . Then  $\Omega$  is Abelian.*

**Proof** Using the fact that diagonal matrices commute,

$$\sigma_u \sigma_v = W^{-1} W \sigma_u W^{-1} W \sigma_v W^{-1} W = W^{-1} W \sigma_v W^{-1} W \sigma_u W^{-1} W = \sigma_v \sigma_u$$

Therefore, the group  $\{\sigma_w : w \in \Omega\}$  of permutation matrices is Abelian. Since  $\Omega$  is isomorphic to this group,  $\Omega$  is also Abelian.  $\square$

Theorem 21 says that if  $\Omega$  is Abelian, then the Fourier transform simultaneously diagonalises the  $\sigma_w$ . Since  $\widehat{\sigma}_w$  is simply the matrix of  $\sigma_w$  with respect to an alternate basis,

one might wonder if there might exist some basis in which the  $\sigma_w$  are diagonal when  $\Omega$  is not Abelian. The message of theorem 22 is that the answer is no. For example, the  $\sigma_w$  corresponding to permutation search spaces (e.g., for the travelling salesman problem) cannot be simultaneously diagonalised.

### 5.5 Examples

**Strings over arbitrary alphabets** The Fourier transform for the case  $\Omega = \mathcal{Z}_3 \times \mathcal{Z}_2$  has already been calculated above. If we take the permutation matrix  $\sigma_{21}$  for example

$$\sigma_{21} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

then a direct calculation gives us

$$\widehat{\sigma}_{21} = \text{diag}(1, -1, -R^*, R^*, -R, R)$$

where  $R = \frac{1}{2}(1 + \sqrt{-3})$ .

**Scheduling jobs on machines** The group  $S_3 \times S_3 \times S_3$  is not Abelian, and therefore, by theorem 22, there is no change of basis in which the permutation matrices associated with structural crossover will simultaneously diagonalise.

## 6 Structural operators for Abelian groups

Just because a crossover operator commutes with an Abelian  $\Omega$ , it is not necessarily the case that it respects the normal subgroups of  $\Omega$ , and so it is not necessarily structural. In this section we consider the special case when crossover and mutation are structural with respect to the usual subgroup decomposition of Abelian  $\Omega$ . This situation corresponds to representing elements of the search space as fixed-length strings (of length  $\ell$ ), with position  $i$  of each string having cardinality  $c_i$ . Crossover and mutation defined in the usual way on such a search space (e.g., one-point, two-point, uniform crossover and bitwise mutation with a given rate) are then examples of structural operators.

In the case where  $\Omega$  is Abelian, there is no need to distinguish the set  $\mathcal{B}$  of binary masks as being separate from  $\Omega$ . Because  $\Omega \cong \mathcal{Z}_{c_0} \times \dots \times \mathcal{Z}_{c_{\ell-1}}$ , the binary masks can be taken to be

$$\mathcal{B} = \{w \in \Omega : w_i \in \{0, 1\}\}$$

Moreover, there is considerably more algebraic structure than in the non Abelian case;  $\Omega$  is a module over the ring  $\mathcal{B}$ , and  $\Omega$  is also a ring in which  $\mathcal{N}$ -schemata are ideals.

**Theorem 23** *Suppose  $\Omega$  is Abelian. If crossover is structural with mixing matrix  $M$ , then  $\widehat{M} = M$ . If mutation is structural with matrix  $U$ , then  $\widehat{U}$  is diagonal.*

**Proof**

$$\begin{aligned}
 \widehat{M}_{u,v} &= \frac{1}{n} \sum_{x,y} \alpha(u \otimes x) \alpha(\ominus v \otimes y) \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [x \otimes b \oplus \bar{b} \otimes y = 0] \\
 &= \frac{1}{n} \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_x \alpha(u \otimes x) [x \otimes b = 0] \sum_y \alpha(\ominus v \otimes y) [\bar{b} \otimes y = 0] \\
 &= \frac{1}{n} \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} \sum_{x \in \Omega_b} \alpha(u \otimes x) \sum_{y \in \Omega_{\bar{b}}} \alpha(\ominus v \otimes y)
 \end{aligned}$$

Appealing to lemma 18 to evaluate the two inner sums (apply the lemma to the choice of search space  $\Omega_b$  and  $\Omega_{\bar{b}}$  respectively), the expression above is

$$\frac{1}{n} \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} |\Omega_b| [u \otimes b = 0] |\Omega_{\bar{b}}| [v \otimes \bar{b} = 0] = \sum_{b \in \mathcal{B}} \frac{\chi_b + \chi_{\bar{b}}}{2} [u \otimes b \oplus v \otimes \bar{b} = 0]$$

Since mutation is structural,  $U_{u,v} = U_{0,v \oplus u}$ . Therefore,

$$\begin{aligned}
 \widehat{U}_{u,v} &= \frac{1}{n} \sum_{x,y} \alpha(u \otimes x) \alpha(\ominus v \otimes y) U_{x,y} = \frac{1}{n} \sum_{x,y} \alpha(u \otimes x) \alpha(\ominus v \otimes y) U_{0,y \oplus x} \sum_w [w = y \oplus x] \\
 &= \frac{1}{n} \sum_x \alpha(u \otimes x) \sum_w U_{0,w} \sum_y \alpha(\ominus v \otimes y) [w = y \oplus x] \\
 &= \frac{1}{n} \sum_x \sum_w \alpha(u \otimes x) \alpha(\ominus v \otimes (w \oplus x)) U_{0,w} = \frac{1}{n} \sum_w \alpha(\ominus v \otimes w) U_{0,w} \sum_x \alpha((u \oplus v) \otimes x) \\
 &= [u = v] \sum_w \alpha(\ominus v \otimes w) U_{0,w}
 \end{aligned}$$

□

This theorem indicates that the mixing scheme corresponding to structural operators acting on an Abelian group are considerably simplified when working in the basis given by the Fourier transform of the group. The permutation matrices are diagonalised, as is the mutation matrix. The mixing matrix corresponding to pure crossover is left unaltered. Using this basis enables the effects of mixing to be calculated much more efficiently.

The effects of the Fourier transform on the selection operator  $\mathcal{F}$  are not so helpful. However, in the case of proportional selection we do have the following interesting result.

**Theorem 24** *Let  $\Omega$  be Abelian with cardinality  $n$  and let  $S = \text{diag}(f)$  be any  $n \times n$  diagonal matrix. Then  $\widehat{S}$  commutes with  $\Omega$ .*

**Proof** Let  $u, v, w \in \Omega$ . Then

$$\begin{aligned}
 \widehat{S}_{u \oplus w, v \oplus w} &= \frac{1}{n} \sum_{x,y} \alpha((u \oplus w) \otimes x) \alpha((\ominus w \oplus v) \otimes y) S_{x,y} = \\
 &= \frac{1}{n} \sum_y \alpha((u \oplus w) \otimes y) \alpha((\ominus w \oplus v) \otimes y) f_y = \\
 &= \frac{1}{n} \sum_y \alpha(u \otimes y) \alpha(w \otimes y) \alpha(\ominus w \otimes y) \alpha(\ominus v \otimes y) f_y = \frac{1}{n} \sum_y \alpha(u \otimes y) \alpha(v \otimes y) f_y = \widehat{S}_{u,v}
 \end{aligned}$$

□

## 7 Conclusion

In this paper we have considered the situation where the search space  $\Omega$  has a group property which can be decomposed as the inner direct sum of a collection of normal subgroups. Structural operators (crossover and mutation) are defined using binary masks over this subgroup decomposition. They are guaranteed to respect all schemata.

The intimate relationship between structural operators and schemata leads us to consider the projection of the search space onto families of competing schemata. Structural operators are shown to have natural well-defined projections on to these schema families. This result, the *Schemata Projection Theorem*, means that the effects of structural crossover and mutation can be calculated independently for any schema family. It is tempting to be carried away by this result and attempt to define a similar projection for selection operators. It has been shown that this is in general impossible to do in any coherent fashion, except for when the population is at a fixed-point. The consequences of this special case have been illustrated with a counterexample to the building block hypothesis.

The Walsh Transform (or Fourier Transform, in its more general form) has been previously shown to be closely connected with the theory of genetic algorithms. In particular, it was already known that in the case of fixed-length  $c$ -ary strings, the transform simultaneously diagonalises the associated set of permutation matrices (Koehler et al., 1997). It is a natural question to ask whether such a transform exists for other search spaces (e.g., the set of permutations required for the travelling salesman problem). We have answered this question exactly: a diagonalising transform exists if and only if the group action with which mixing commutes is itself commutative. In particular, if we identify the travelling salesman problem with the group of all permutations of the set of cities, such a transform does not exist, since the group is non-commutative.

### Appendix: Useful algebraic identities

It is easy to check that for all  $u, v \in \Omega$  and  $a, b \in \mathcal{B}$

$$\begin{array}{ll}
 u = u \otimes \mathbf{1} & \mathbf{1} = b \oplus \bar{b} \\
 0 = b \oplus b & 0 = b \otimes \bar{b} \\
 (u \oplus v) \otimes b = (u \otimes b) \oplus (v \otimes b) & (u \otimes b) \oplus (v \otimes \bar{b}) = (v \otimes \bar{b}) \oplus (u \otimes b) \\
 (u \otimes b) \otimes a = u \otimes (b \otimes a) & \ominus(b \otimes u) = b \otimes (\ominus u) \\
 u \otimes b \oplus \Omega_b = \Omega_b & u \ominus \Omega_b = u \oplus \Omega_b \\
 \Omega = \{u \oplus v : u \in \Omega_b, v \in \Omega_{\bar{b}}\} & u \otimes \bar{b} = 0 \Leftrightarrow u \in \Omega_b
 \end{array}$$

### References

- Goldberg, D. E. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning*. Addison Wesley, Reading, MA.
- Holland, J. (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, Michigan.
- Koehler, Bhattacharyya, and Vose (1997). General cardinality genetic algorithms. *Evolutionary Computation*, 5(4):439–459.
- Lang, S. (1993). *Algebra*. Addison-Wesley, Reading, MA, third edition.
- Radcliffe, N. J. (1992). The algebra of genetic algorithms. *Anal of Mathematics and Artificial Intelligence*, 10:339–384.



- Rowe, J. E., Vose, M. D., and Wright, A. H. (2002). Group properties of crossover and mutation. *Evolutionary Computation*, 10(2):151–184.
- Stephens, C. and Waelbroeck, H. (1999). Schemata evolution and building blocks. *Evolutionary Computation*, 7(2).
- Vose, M. D. (1991). Determining concepts by group membership. *Applicable Algebra in Engineering, Communication and Computing*, 2:139–146.
- Vose, M. D. (1999). *The Simple Genetic Algorithm: Foundations and Theory*. MIT Press, Cambridge, MA.
- Vose, M. D. and Wright, A. H. (1998). The simple genetic algorithm and the Walsh transform: Part I: Theory. *Evolutionary Computation*, 6(3):253–273.
- Vose, M. D. and Wright, A. H. (2001). Form invariance and implicit parallelism. *Evolutionary Computation*, 9(3):355–370.
- Wright, A. H., Vose, M. D., and Rowe, J. E. (2003). Implicit parallelism. In Foster, J., editor, *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO) 2003*. Springer.