On the Search Biases of Homologous Crossover in Linear Genetic Programming and Variable-length Genetic Algorithms

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

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

Alden H. Wright
Computer Science Department
University of Montana, USA
wright@cs.umt.edu

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

Abstract

With a schema-theoretic approach and experiments we study the search biases produced by GP/GA homologous crossovers when applied to linear, variable-length representations. By specialising the schema theory for homologous crossovers we show that these operators are unbiased with respect to string length. Then, we provide a fixed point for the schema evolution equations where the population presents a statistically independent distribution of primitives. This is an important step towards generalising Geiringer's theorem and the notion of linkage equilibrium.

1 INTRODUCTION

Search algorithms typically include three main steps which are iterated in succession: choosing one or multiple points of the search space the neighbourhood of which to explore further, applying *expansion operators* to obtain a new set of points, deciding what to do with the points previously visited and with the newly generated ones. For example, in the case of genetic algorithms and genetic programming, selection corresponds to the first task, crossover and mutation are the expansion operators, and the replacement strategy corresponds to the third task. Note that many more steps may be included if one looks at search algorithms at a finer level of abstraction (Poli & Logan, 1996), but this is not particularly important for the purposes of this discussion. What is important is that different algorithms will use different strategies to realise the different steps. This leads to the sampling of the search space according to different schedules. With the exception of random search, this means that some areas of the search space will be explored sooner, will be allocated more samples, or will be ignored altogether. This is what we mean by *search bias*.

Clearly the bias of an algorithm is the result of the interaction of the biases of all its components. In the case of fixed length genetic algorithms, a lot of attention has been devoted to the biases of all such components: selection, mutation and crossover, and replacement. Some of the resulting studies apply also to the case of variable-length-structure evolution. For example, the focusing effects of selection will be exactly the same, since selection is representation-independent. However, very little is known about the biases of the genetic operators used to evolve variable length representations, such as those used in linear GP (Nordin, 1994; O'Neill & Ryan, 2001) or in variable-length GAs.

Knowing the biases introduced by the operators is very important, since it leads to a deeper understanding of the search algorithm under investigation. This, in turn, allows an informed choice of operators, parameter settings and even initialisation strategies for particular problems. How can one investigate these biases? One possibility is to use carefully designed empirical studies. In the past these have shed some light on the internal dynamics of GP (e.g. on bloat (McPhee & Miller, 1995; Soule *et al.*, 1996; Langdon *et al.*, 1999)), but rarely has the evidence been general and conclusive. This is because these studies can only consider a limited number of benchmark problems, and so it is impossible to know whether, and to which extent, the observed behaviour is applicable to other problems. An alternative is to perform theoretical studies. Often these may lead to more general and precise conclusions, but they are definitely much harder and slower to carry out. Also, sometimes the complexity of the mathematics involved in these studies forces the researcher to make simplifying hypotheses which may limit the explanatory power of the results.

A class of recent theoretical results which require very few, if any, simplifications goes under the name of *exact schema theorems*.[1] These provide probabilistic models (the schema evolution equations) of the expected behaviour of a GA or a GP system over one, or, under certain assumptions, multiple generations (Poli, 2001*b*; Langdon & Poli,

[1] The word "exact" refers to the fact that, unlike earlier results, these theorems provide an exact value, rather than a lower bound, for the expected number of individuals in a schema in the next generation.

2002). The main advantage of these exact models is that they provide a natural way of coarse graining the huge number of degrees of freedom present in a genetic algorithm (Stephens & Waelbroeck, 1997). Exact schema theorems have recently become available for fixed-length GAs with one-point crossover and mutation (Stephens & Waelbroeck, 1997; Stephens & Waelbroeck, 1999), and general homologous crossover and mutation (Vose & Wright, 2001; Stephens, 2001). Even more recent is the development of exact schema theorems for variable-length GAs, linear GP and tree-based GP. These cover a variety of crossover and mutation operators including one-point crossover (Poli, 2000; Poli, 2001*b*), subtree-swapping crossovers (Poli, 2001*a*; Poli & McPhee, 2001*b*; McPhee & Poli, 2001), different types of subtree mutation and headless chicken crossover (Poli & McPhee, 2001*a*; McPhee *et al.*, 2001), and homologous crossovers (Poli & McPhee, 2001*c*).

These exact models can be used to understand an evolutionary system and study its behaviour in two different ways. This can be done either through simulation (i.e., by numerically iterating the equations) or through mathematical analysis. Although exact GP schema equations have become available only very recently, early studies indicate their usefulness, for example, in providing a deeper understanding of emergent phenomena such as bloat (Poli & McPhee, 2001*b*; McPhee & Poli, 2001). Also, in general, as indicated above the availability of exact models for different operators allows a formal study of the biases of those operators. Steps forward in this direction have recently been made in (Poli & McPhee, 2001*b*; McPhee & Poli, 2001; McPhee *et al.*, 2001), where a class of Gamma program-length distributions has been shown to represent a natural attractor for variable-length linear systems under GP subtree crossover and in (Poli *et al.*, 2002) where we have extended the study to other biases of subtree crossover.

In this paper we study the biases of the whole class of homologous GP crossover operators for the case of linear GP and variable-length GAs. These are a set of operators, including GP one-point crossover (Poli & Langdon, 1997) and GP uniform crossover (Poli & Langdon, 1998*a*), where the offspring are created preserving the position of the genetic material taken from the parents. These operators are important because they are the natural generalisation of the corresponding GA operators. So, the theory presented here is a generalisation of corresponding GA theory.

The paper is organised as follows. We start by providing some background information on the exact GP schema theory for homologous crossover and Geiringer's theorem in Sections 2 and 3. Then, we simplify the theory for the case of linear, but variable-length, structures in Section 4, and show that homologous crossover is totally unbiased with respect to string length (Section 5). In Section 6 we provide a fixed point for the schema evolution equations which is a first step towards generalising Geiringer's theorem (Geiringer, 1944) and the notion of linkage equilibrium, which, until now, were applicable only to fixed-length representations. The fixed point and some experimental evidence (reported in Section 7) indicate the presence of a bias which pushes the population towards a statistically independent distributions of primitives, as discussed in Section 8, where we also draw some conclusions.

2 SCHEMA THEORY BACKGROUND

Schemata are sets of points in a search space sharing some syntactic feature. For example, for GAs operating on binary strings the syntactic representation of a schema is usually a string of symbols from the alphabet $\{0,1,*\}$, where the character * is interpreted as a "don't care" symbol. Typically schema theorems are descriptions of how the number of members of the population belonging to a schema vary over time. If $\alpha(H,t)$ denotes the probability that at time t a newly created individual samples (or matches) the schema H, which we term the *total transmission probability* of H, then an exact schema theorem for a generational system is simply

$$E[m(H,t+1)] = M\,\alpha(H,t),$$

where M is the population size, $m(H,t+1)$ is the number of individuals sampling H at generation $t+1$ and $E[\cdot]$ is the expectation operator. Holland's (Holland, 1975) and other (e.g. (Poli & Langdon, 1998*b*)) worst-case-scenario schema theories normally provide a lower bound for $\alpha(H,t)$ or, equivalently, for $E[m(H,t+1)]$. However, recently exact schema theorems (Stephens & Waelbroeck, 1997; Stephens & Waelbroeck, 1999; Poli, 2000; Poli, 2001*b*; Poli, 2001*a*; Poli & McPhee, 2001*b*; McPhee & Poli, 2001; Poli & McPhee, 2001*a*; McPhee *et al.*, 2001; Stephens, 2001; Poli & McPhee, 2001*c*) which provide the exact value for $\alpha(H,t)$ have become available for GAs and GP with a variety of operators. In the remainder of this section we will introduce the various elements which are necessary to understand the exact schema equations for GP homologous crossover, since these will be the starting point for the new results in this paper.

Let us start from our definition of schema for GP. Syntactically a GP schema is a tree composed of functions from the set $\mathcal{F}\cup\{=\}$ and terminals from the set $\mathcal{T}\cup\{=\}$, where $\mathcal{F}$ and $\mathcal{T}$ are the function and terminal sets used in a GP run. The primitive $=$ is a "don't care" symbol which stands for a *single* terminal or function. A schema H represents the set of all programs having the same shape as H and the same non-$=$ nodes as H. Particularly important for the GP schema theory are schemata containing "don't care" symbols only, since they represent all the programs of a particular shape. Let G_1, G_2, $\cdots$ be an enumeration of such shape-representing schemata.

In GP homologous crossovers the offspring are created by exchanging genetic material (nodes and subtrees) taken

from the same position in the parents trees. To account for the possible structural diversity of the two parents, the selection of the nodes and the roots of the subtrees to swap are constrained to belong to the *common region*. This is the largest rooted region where the two parent trees have the same topology.

In order to define more precisely how GP homologous crossovers work, we start by providing a formal definition of common region. The common region between two generic trees h_1 and h_2 is the set $C(h_1, h_2) = \{(d, i)|\mathcal{C}(d, i, h_1, h_2)\}$, where (d, i) is a pair of coordinates in a Cartesian node reference system (Poli, 2001*a*; Poli & McPhee, 2001*c*). The predicate $\mathcal{C}(d, i, h_1, h_2)$ is true if $(d, i) = (0, 0)$ (i.e., if (d, i) is the root node). It also true if $A(d-1, i', h_1) = A(d-1, i', h_2) \neq 0$ and $\mathcal{C}(d-1, i', h_1, h_2)$ is true, where $A(d, i, h)$ returns the arity of the node at coordinates (d, i) in h, $i' = \lfloor i/a_{\max} \rfloor$, $a_{\max}$ is the maximum arity of the functions in the function set, and $\lfloor \cdot \rfloor$ is the integer-part function. The predicate is false otherwise. The notion of common region can be applied to schemata, too.

To complete our formal description of the class of GP homologous crossovers, we need to extend to GP the notions of crossover masks and recombination distributions used in genetics (Geiringer, 1944) and in the GA literature (Booker, 1992; Altenberg, 1995; Spears, 2000). Let us first briefly recall the definition of these notions for a GA operating on fixed-length binary strings. In this case a crossover mask is simply a binary string. When crossover is executed, the bits of the offspring corresponding to the 1's in the mask will be taken from one parent, those corresponding to 0's from the other parent. If the GA operates on strings of length N, then 2^N different crossover masks are possible. If, for each mask i, one defines a probability, p_i, that the mask is selected for crossover, then it is easy to see how different crossover operators can simply be interpreted as different ways of choosing the probability distribution p_i. The distribution p_i is called a *recombination distribution*.

For the more general case of GP and variable-length GAs, for any given common region c we can define a set of *GP crossover masks*, χ_c, which contains all different trees with the same size and shape as c which can be built with nodes labelled 0 and 1 (Poli & McPhee, 2001*c*; Poli *et al.*, 2001). Each crossover mask represents one of the ways in which one could generate an offspring through crossover: nodes of the offspring corresponding to internal 1's in the mask will be taken from the first parent, nodes corresponding to internal 0's from the second parent, subtrees of the first parent whose root corresponds to leaves labelled with a 1 in the mask will be transferred to the same position in the offspring, and, finally, subtrees of the second parent whose root corresponds to leaves labelled with a 0 in the mask will be transferred to the same position in the offspring. The *GP recombination distribution* p_l^c gives the probability that, for a given common region c, crossover mask l will be chosen from the set χ_c. Each GP homologous crossover is characterised by a different recombination distribution. Since the size and shape of the common region can be inferred from the mask l, in the following we will often omit the superscript c from p_l^c.

Finally, before we introduce the exact schema equation for GP homologous crossover developed in (Poli & McPhee, 2001*c*) we need to define the notion of hyperschema. A *GP hyperschema* is a rooted tree composed of internal nodes from $\mathcal{F} \cup \{=\}$ and leaves from $\mathcal{T} \cup \{=, \#\}$. Again, = is a "don't care" symbols which stands for exactly one node, while # stands for any valid subtree. In the theory we use hyperschemata to represent the characteristics the parents must have to produce instances of a particular schema of interest.

The exact schema equations for GP with homologous crossover are

$$\alpha(H, t) = (1 - p_{xo}) p(H, t) + p_{xo} \alpha_{xo}(H, t) \qquad (1)$$

where

$$\alpha_{xo}(H, t) = \sum_j \sum_k \sum_{l \in \chi_{C(G_j, G_k)}} p_l \; p(\Gamma(H, l) \cap G_j, t) p(\Gamma(H, \bar{l}) \cap G_k, t), \qquad (2)$$

p_{xo} is the crossover probability, $p(H, t)$ is the selection probability of the schema H and $\bar{l}$ is the complement of the GP crossover mask l (i.e. it is a tree with the same structure as l but with the 0's and 1's swapped). $\Gamma(H, l)$ is defined to be the empty set if l contains any node not in H. Otherwise it is the hyperschema obtained by replacing certain nodes in H with either = or # nodes: (1) if a node in H corresponds to (i.e., has the same coordinates as) a non-leaf node in l that is labelled with a 0, then that node in H is replaced with a =, (2) if a node in H corresponds to a leaf node in l that is labelled with a 0, then it is replaced with a #, (3) all other nodes in H are left unchanged.

As discussed in (Poli & McPhee, 2001*c*), it is possible to show that, in the absence of mutation, Equations 1 and 2 generalise and refine not only approximate GA and GP schema theorems (Holland, 1975; Poli & Langdon, 1997; Poli & Langdon, 1998*b*) but also more recent exact schema theorems (Stephens & Waelbroeck, 1997; Stephens & Waelbroeck, 1999; Poli, 2000; Stephens, 2001).

In the following we will use a slightly different form for Equation 2 which exploits the symmetries in the process of selection of the parent programs. This can be obtained by dividing each set of crossover masks $\chi_{C(G_j, G_k)}$ into two non-overlapping sets $\chi'_{C(G_j, G_k)}$ and $\bar{\chi}'_{C(G_j, G_k)}$ such that for each mask $x \in \chi'_{C(G_j, G_k)}$, there is a mask $y \in \bar{\chi}'_{C(G_j, G_k)}$ such that $y = \bar{x}$, and *vice versa*. Then, by reordering the terms, it is easy to prove that:

Theorem 1.

$$\alpha_{xo}(H,t) = \sum_j \sum_k \sum_{l \in \chi'_{C(G_j,G_k)}} (p_l + p_{\bar{l}}) \quad (3)$$
$$p(\Gamma(H,l) \cap G_j, t) p(\Gamma(H,\bar{l}) \cap G_k, t)$$

3 GEIRINGER'S THEOREM

In this section we briefly introduce Geiringer's theorem (Geiringer, 1944), an important result with implications both for natural population genetics and evolutionary algorithms (Booker, 1992; Spears, 2000). Geiringer's theorem indicates that, in a population of fixed-length chromosomes repeatedly undergoing crossover (in the absence of mutation and selective pressure), the probability of finding a generic string $h_1 h_2 \cdots h_N$ approaches a limit distribution which is only dependent on the distribution of the alleles h_1, h_2, etc. in the initial generation. More precisely, if $\Phi(h_1 h_2 \cdots h_N, t)$ is the proportion of individuals of type $h_1 h_2 \cdots h_N$ at generation t and $\Phi(h_i, t)$ is the proportion of individuals carrying allele h_i then

$$\lim_{t \to \infty} \Phi(h_1 h_2 \cdots h_N, t) = \prod_{i=1}^{N} \Phi(h_i, 0). \quad (4)$$

This result is valid for all homologous crossover operators which allow any two loci to be separated by recombination. Strictly speaking the result is valid only for infinite populations.

If one interprets $\Phi(h_1 h_2 \cdots h_N, t)$ as a probability distribution of the possible strings in the population, we can interpret Equation 4 as saying that such a distribution is converging towards independence. When, at a particular generation t, the frequency of any string in a population $\Phi(h_1 h_2 \cdots h_N, t)$ equals $\prod_{i=1}^{N} \Phi(h_i, t)$, the population is said to be in *linkage equilibrium* or *Robbins' proportions*.

It is trivial to generalise Geiringer's theorem to obtain the expected fixed-point proportion of a generic linear fixed-length GA schema H for a population undergoing crossover only:

$$\lim_{t \to \infty} \Phi(H, t) = \prod_{i \in \Delta(H)} \Phi(*^{i-1} h_i *^{N-i}, 0), \quad (5)$$

where $\Delta(H)$ is the set of indices of the defining symbols in H, h_i is one such defining symbols and we used the power notation x^y to mean x repeated y times. (Note that $\Phi(*^{i-1} h_i *^{N-i}, t)$ coincides with $\Phi(h_i, t)$.)

4 EXACT SCHEMA THEORY FOR LINEAR STRUCTURES

As indicated in Section 1 in this paper we will consider the biases of the homologous crossovers in the case of variable-size linear representations. We start by specialising Equation 3 to this case.

When only unary functions are used in tree-based GP, schemata (and programs) can only take the form $(h_1(h_2(h_3....(h_{N-1}h_N)....)))$ where $N > 0$, $h_i \in \mathcal{F} \cup \{=\}$ for $1 \leq i < N$, and $h_N \in \mathcal{T} \cup \{=\}$. Therefore, they can be written unambiguously as strings of symbols of the form $h_1 h_2 h_3 .... h_{N-1} h_N$. It should be noted that these strings of symbols do not have to be necessarily interpreted as representing programs. If one uses a special terminal set $\mathcal{T}$ including only one terminal, say `EOR` (for End Of Representation), which will be ignored when the representation is interpreted, then strings of the form $h_1 h_2 h_3 .... h_{N-1} h_N$ can be interpreted as chromosomes of length $N-1$ (since h_N can only be `EOR`). So, if $\mathcal{F} = \{0, 1\}$, where 0 and 1 are "unary functions",[2] our GP system will explore the space of variable length binary strings. If instead $\mathcal{F}$ includes the "unary functions" $\{$`ADD R0 R1`, `MUL R0 R1`, $\ldots\}$, then our tree-based GP system explores the same search space as a machine-code GP system with the same primitive set (Nordin & Banzhaf, 1995). So, in general our specialisation of Equation 3 will be valid for variable-length GAs and linear GP.

In the specialisation to the linear case we replace the "don't care" symbol "=" with the more standard symbol "*". Also, as we did previously, we represent repeated symbols in a string using the power notation. Since in this case all trees are linear, the space of program shapes can be enumerated by $\{G_n\}$ where G_n is $*^n$ for $n > 0$. Given this, the common region between shapes G_j and G_k is simply the shorter of the two schemata, i.e. $C(G_j, G_k) = G_{j \downarrow k} = *^{j \downarrow k}$ where the operator $\downarrow$ returns the minimum of its two arguments. Therefore, the set of crossover masks in the common region, $\chi_{C(G_j,G_k)} = \chi_{*^{j \downarrow k}}$, can be identified with the set $\{0,1\}^{j \downarrow k}$. Below we will use $N(l)$ to denote the length of mask l, and the notation l_i to indicate the i-th element of bitmask l. We will also use the operators

$$a \bullet b = \begin{cases} a & \text{if } b = 1 \\ * & \text{otherwise,} \end{cases}$$

$$a_1 a_2 \cdots \circ b = \begin{cases} a_1 a_2 \cdots & \text{if } b = 1 \\ \# & \text{otherwise,} \end{cases}$$

where a and b are bits, $a_1 a_2 \cdots$ is a bit string and # stands for any sequence of at least one primitive. With this notation, it is easy to show that, in a linear representation, if $N(l) > N$ then $\Gamma(H, l)$ is the empty set and $\Gamma(H, l) \cap G_j = \emptyset \cap *^j = \emptyset$. If $N(l) \leq N$, $\Gamma(H, l)$ is

$$(h_1 \bullet l_1) \cdots (h_{N(l)-1} \bullet l_{N(l)-1})(h_{N(l)} \cdots h_N \circ l_{N(l)})$$

and

[2] We are not interested in the output of these functions, but simply in their topological organisation within the individual.

$$\Gamma(H,l)\cap G_j = \Gamma(H,l)\cap *^j = \tag{6}$$
$$\begin{cases} (h_1 \bullet l_1)\cdots(h_{N(l)-1}\bullet l_{N(l)-1})*^{j-N(l)+1} \\ \qquad\qquad \text{if } j \geq N(l) \text{ and } l_{N(l)}=0, \\ (h_1 \bullet l_1)\cdots(h_{N(l)-1}\bullet l_{N(l)-1})h_{N(l)}\cdots h_N \\ \qquad\qquad \text{if } j = N \text{ and } l_{N(l)}=1, \\ \emptyset \qquad\qquad \text{otherwise.} \end{cases}$$

Thus, $p(\Gamma(H,l)\cap G_j,t) = 0$ for all $j \neq N$ for all the masks l for which $l_{N(l)}=1$. So, if we choose $\chi'_{C(G_j,G_k)} = \{0,1\}^{j\downarrow k-1}\times\{1\}$, in Equation 3 only the terms for $j=N$ can be non-zero. Using this simplification and the previous results, one can transform Equation 3 into:

Theorem 2. *If* $\chi'_{*^{N\downarrow k}} = \{0,1\}^{j\downarrow k-1}\times\{1\}$*, then*

$$\alpha_{xo}(h_1...h_N,t) = \sum_{k>0}\sum_{l\in\chi'_{*^{N\downarrow k}}}(p_l+p_{\bar{l}}) \tag{7}$$
$$p((h_1\bullet l_1)\cdots(h_{N\downarrow k-1}\bullet l_{N\downarrow k-1})h_{N\downarrow k}\cdots h_N,t)$$
$$p((h_1\bullet \bar{l}_1)\cdots(h_{N\downarrow k-1}\bullet \bar{l}_{N\downarrow k-1})*^{k-N\downarrow k+1},t).$$

5 LENGTH EVOLUTION

Equation 7 can be used to study, among other things, the evolution of size in linear GP/GA systems. This is because it can be specialised to schemata of the form $*^N$ obtaining:

$$\begin{aligned}\alpha_{xo}(*^N,t) &= \sum_{k>0}\sum_{l\in\chi'_{*^{N\downarrow k}}}(p_l+p_{\bar{l}})\,p(*^N,t)p(*^k,t)\\ &= p(*^N,t)\sum_{k>0}p(*^k,t)\sum_{l\in\chi'_{*^{N\downarrow k}}}(p_l+p_{\bar{l}})\,.\end{aligned}$$

But $\sum_{l\in\chi'_{*^{N\downarrow k}}}(p_l+p_{\bar{l}})=1$ and $\sum_{k>0}p(*^k,t)=1$, so:

Theorem 3.
$$\alpha_{xo}(*^N,t) = p(*^N,t). \tag{8}$$

This result indicates that in linear representations length evolves under homologous crossovers as if selection only was acting. So, homologous crossovers are totally unbiased with respect to program length. The lack of length bias of homologous crossovers is made particularly clear if one assumes a flat fitness landscape in which $p(H,t)=\Phi(H,t)$ for all H. In these conditions all the dynamics in the system must be caused by crossover or by sampling effects. In the infinite population limit, the total transmission probability $\alpha(H,t)$ can also be interpreted as the proportion of individuals in the population in H at generation $t+1$, $\Phi(H,t+1)$. So, for an infinite population and a flat landscape Equation 8 becomes $\Phi(*^N,t+1)=\Phi(*^N,t)$, whereby

Corollary 4. *For a flat landscape, an infinite population and any* $t>0$

$$\Phi(*^N,t) = \Phi(*^N,0).$$

This equation is important because it shows that when a homologous crossover alone is acting, any initial distribution of lengths is a fixed point length distribution for the system.

6 EXTENSION OF GEIRINGER'S THEOREM

A full extension of Geiringer's theorem to linear, variable-length structures and homologous GP crossover would require two steps: (a) proving that, in the absence of mutation and of selective pressure and for an infinite population, a distribution $\Phi(h_1h_2\cdots h_N,t)$, where the alleles/primitives can be considered independent stochastic variables, is a fixed point, and (b) showing that the system indeed moves towards that fixed point. In this paper we prove (a) mathematically and provide experimental evidence for (b).

Theorem 5. *A fixed point distribution for the proportion of a linear, variable-length schema* $h_1h_2\cdots h_N$ *under homologous crossover for an infinite population on a flat fitness landscape in the absence of mutation is*

$$\Phi(h_1h_2\cdots h_N,t) = \Phi(*^{N-1}h_N,0)\prod_{i=1}^{N-1}\frac{\Phi(*^{i-1}h_i\#,0)}{\Phi(*^i\#,0)}, \tag{9}$$

where

$$\Phi(*^{i-1}h_i\#,0) = \sum_{n>0}\Phi(*^{i-1}h_i*^n,0)$$

and

$$\Phi(*^i\#,0) = \sum_{n>0}\Phi(*^{i+n},0).$$

Proof. Since the fitness landscape is flat, $p(H,t) = \Phi(H,t)$ for any schema. Also, because the population is infinite, $\alpha(H,t)=\Phi(H,t+1)$. So, combining Equations 1 and 7 yields

$$\begin{aligned}&\Phi(h_1...h_N,t+1) \qquad (10)\\ &= (1-p_{xo})\Phi(h_1...h_N,t) + p_{xo}\sum_{k>0}\sum_{l\in\chi'_{*^{N\downarrow k}}}(p_l+p_{\bar{l}})\\ &\times\ \Phi((h_1\bullet l_1)\cdots(h_{N\downarrow k-1}\bullet l_{N\downarrow k-1})h_{N\downarrow k}\cdots h_N,t)\\ &\times\ \Phi((h_1\bullet \bar{l}_1)\cdots(h_{N\downarrow k-1}\bullet \bar{l}_{N\downarrow k-1})*^{k-N\downarrow k+1},t).\end{aligned}$$

We prove that Equation 9 represents a fixed point for Equation 10 by substituting the former in the latter and reordering the terms, obtaining

$$\begin{aligned}&\Phi(h_1...h_N,t+1)\\ &= (1-p_{xo})\Phi(*^{N-1}h_N,0)\prod_{i=1}^{N-1}\frac{\Phi(*^{i-1}h_i\#,0)}{\Phi(*^i\#,0)}\\ &+ p_{xo}\Phi(*^{N-1}h_N,0)\sum_{k>0}\Phi(*^k,0)\end{aligned}$$

$$\times \sum_{l \in \chi'_{*^{N\downarrow k}}} (p_l + p_{\bar{l}})$$
$$\times \prod_{i=1}^{N\downarrow k-1} \left(\frac{\Phi(*^{i-1}(h_i \bullet l_i)\#, 0)\Phi(*^{i-1}(h_i \bullet \bar{l}_i)\#, 0)}{(\Phi(*^i\#, 0))^2} \right)$$
$$\times \prod_{i=N\downarrow k}^{N-1} \frac{\Phi(*^{i-1}h_i\#, 0)}{\Phi(*^i\#, 0)}.$$

Whatever the value of bit l_i in the mask, either $h_i \bullet l_i = h_i$ and $h_i \bullet \bar{l}_i = *$, or $h_i \bullet l_i = *$ and $h_i \bullet \bar{l}_i = h_i$. In either case $\Phi(*^{i-1}(h_i \bullet l_i)\#, 0)\Phi(*^{i-1}(h_i \bullet \bar{l}_i)\#, 0) = \Phi(*^{i-1}h_i\#, 0)\Phi(*^i\#, 0)$. So, after reordering the terms, we obtain:

$$\begin{aligned}
&\Phi(h_1 ... h_N, t+1) \\
&= (1 - p_{xo})\Phi(*^{N-1}h_N, 0) \prod_{i=1}^{N-1} \frac{\Phi(*^{i-1}h_i\#, 0)}{\Phi(*^i\#, 0)} \\
&+ p_{xo}\Phi(*^{N-1}h_N, 0) \prod_{i=1}^{N-1} \frac{\Phi(*^{i-1}h_i\#, 0)}{\Phi(*^i\#, 0)} \\
&\times \sum_{k>0} \Phi(*^k, 0) \sum_{l \in \chi'_{*^{N\downarrow k}}} (p_l + p_{\bar{l}}) \\
&= \Phi(*^{N-1}h_N, 0) \prod_{i=1}^{N-1} \frac{\Phi(*^{i-1}h_i\#, 0)}{\Phi(*^i\#, 0)} \qquad \square
\end{aligned}$$

It is important to note that, although Equation 9 provides a family of fixed points, this does not prove rigorously that any population will always converge to one of them. Proving this is complex and requires much more space than is available for this conference. We will provide the proof in a future more extended publication. Instead, in the following section we will describe experimental results which strongly suggest that indeed populations move toward an independent allele/primitive distribution.

7 EXPERIMENTAL RESULTS

In order to check the theoretical results in this paper we set up a population of variable length strings consisting of 1,000,000 individuals. All individuals had the same terminal allele, 0, while two types of non-terminal alleles were used: alleles of type 0 and alleles of type 1. The majority of alleles were of type 0 and represented a "background" against which alleles of type 1 (the "contrast medium" we used to study the dynamics of non-terminal alleles) could be more easily traced. Initially, alleles of type 1 occupied all the non-terminal loci of strings of a given length only (which was varied between experiments). The terminal locus of those strings was occupied by an allele of type 0. All other terminal and non-terminal alleles in the population were of type 0.

In our experiments we used two different initial length distributions: a Gamma distribution with mean 10.5, and a uniform distribution with lengths between 1 and 20. Each population was run for 100 generations. The system was a generational GP/GA system with either one-point crossover or uniform crossover (applied with 100% probability) and a flat fitness landscape. One-point crossover is a homologous crossover operator which, for variable length strings, is characterised by the recombination distribution

$$p_l^{*^n} = \begin{cases} 1/n & \text{if } l \in \{0^n, 10^{n-1}, 110^{n-2}, \ldots, 1^{n-1}0\}, \\ 0 & \text{otherwise.} \end{cases}$$

Uniform crossover has the recombination distribution $p_l^{*^n} = 2^{-n}$.

Multiple independent runs were not required since the population size was sufficiently large to remove any significant statistical variability and therefore to approximate the infinite-population behaviour (for each program length we had tens of thousands of individuals on average).

We start by checking what happens to the length distribution over time. Figure 1 shows that the distribution of program length is at a fixed point when the population is initialised using either a uniform length distribution or a discrete Gamma distribution. This corroborates our finding that any length distribution is a fixed point (Corollary 4). Note that the small variations in the plots are due to genetic drift (i.e. a finite population effect).

Let us now consider the allele dynamics. Figure 2 shows how the distribution of alleles of type 1 varies over a number of generations in a population initialised with the uniform length distribution. In Figure 2(a) only strings of length 2 included alleles of type 1 (in locus 1). However, under one-point crossover within a few generations (see Figures 2(b) and (c)) the relative proportion of strings with a 1 in locus 1 reached the equilibrium value predicted by applying Equation 9 to the schema $1*^{N-1}$:

$$\begin{aligned}
\lim_{t\to\infty} \frac{\Phi(1*^{N-1}, t)}{\Phi(*^N, 0)} &= \frac{\Phi(1\#, 0)}{\Phi(*\#, 0)} \prod_{i=2}^{N-1} \frac{\Phi(*^{i-1} * \#, 0)}{\Phi(*^i\#, 0)} \\
&\approx \frac{\frac{1}{20}}{\frac{19}{20}} \times \prod_{i=2}^{N-1} 1 = \frac{1}{19} \approx 0.0526.
\end{aligned}$$

The same asymptotic value is approached when only strings of length 10 included alleles of type 1 at generation 0, as shown in Figures 2(d)–(g). Qualitatively the behaviour of other loci is the same, but the asymptotic values reached are slightly different, which is predicted by Theorem 5.

Figure 2 reveals that the speed with which alleles in different loci approach their asymptotic value varies. While alleles in locus 1 move quickly towards their fixed point, the convergence speed decreases as the locus position in-

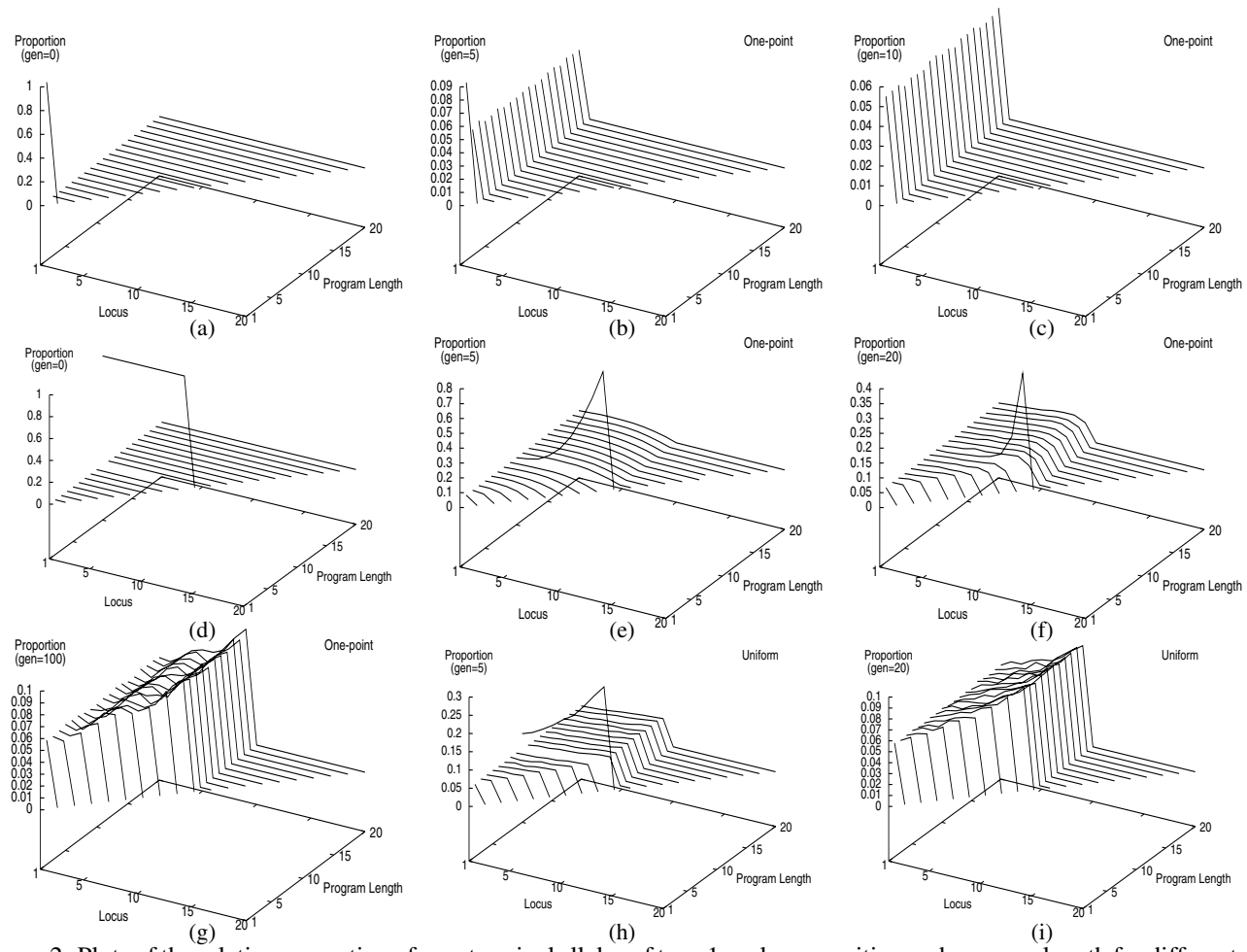

Figure 2: Plots of the relative proportion of non-terminal alleles of type 1 vs. locus position and program length for different generations under different crossover operators. The population was initialised with a uniform length distribution, where only programs of length ℓ contained non-terminal alleles of type 1. The value of ℓ was 2 in (a)–(c) and 10 in (d)–(i). One-point crossover was used in (b), (c), (e), (f) and (g), uniform crossover in (h) and (i).

creases. This is due to the fact that alleles occupying non-terminal loci that are present in a large number of individuals will be swapped more frequently than alleles occupying loci present in a small number of individuals.

The behaviour of uniform crossover is almost identical. Analysis of our results revealed that the only difference is the speed with which alleles in different loci approach their asymptotic value. Uniform crossover mixes alleles more quickly as can easily be seen, for example, by comparing Figures 2(e) and (f) with Figures 2(h) and (i), respectively.

To further verify that under homologous crossover the population tends towards an independent allele distribution, we performed an experiment with exactly the same set up as in Figures 2(h) and (i) but this time we kept track of the co-occurrence of pairs of non-terminal alleles within the class of programs of length 10. So, for each generation we obtained a set of four 9×9 co-occurrence frequency matrices, one for each possible choice of a pair of the non-terminal alleles 0 and 1. An element at position (r, c) of the co-occurrence matrix for non-terminal alleles a and b, represented the average number of times allele a was present in locus r while at the same time allele b was present in locus c in strings of length 10. Once normalised by the total number of strings of length 10, the diagonal elements of the 0/0 and 1/1 matrices represent the proportions of alleles of type 0 and 1, respectively, present at each locus.

For all allele pairs the co-occurrence matrices tended to those predicted by the theory. For example, for the allele pair 0/0 the theoretical values for the off-diagonal elements can be calculated using the following equation (obtained from Equation 9)

$$\lim_{t \to \infty} \frac{\Phi(*^a 0 *^b 0 *^{10-a-b-2}, t)}{\Phi(*^{10}, 0)}$$

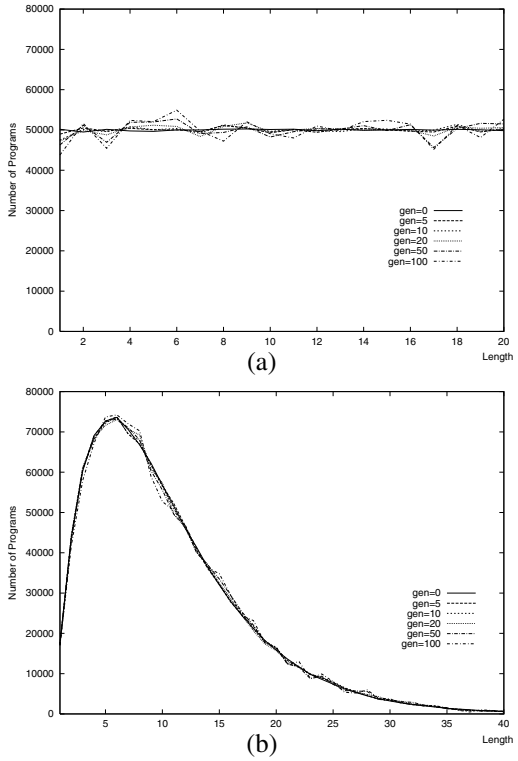

Figure 1: Plots of the number of programs vs. program length for different generations for populations initialised with a uniform (a) and a Gamma (b) length distribution.

$$= \frac{\Phi(*^a 0\#, 0)}{\Phi(*^{a+1}\#, 0)} \times \frac{\Phi(*^{a+b+1}0\#, 0)}{\Phi(*^{a+b+2}\#, 0)}$$
$$\approx \frac{\frac{18-a}{20}}{\frac{19-a}{20}} \times \frac{\frac{17-a-b}{20}}{\frac{18-a-b}{20}} = \frac{18-a}{19-a} \times \frac{17-a-b}{18-a-b}.$$

These values are extremely close to the frequency of 0/0 co-occurrence measured at generation 100 in our runs, the root mean square error between the above-diagonal elements being 0.004 (note that the co-occurrence matrix is symmetric). This is a tiny error considering that all frequencies were bigger than 0.8.

8 DISCUSSION AND CONCLUSIONS

Characterisations of the genetic biases of the operators (such as the ones offered in this paper and in (Poli *et al.*, 2002)) are important because they allow the users of GP/GA systems to evaluate whether their operators provide the desired search behaviour for the system. If this is not the case, then the knowledge of the search biases of other operators allows for an informed choice for an alternative.

Here we have focused our attention on the biases of homologous crossovers with respect to length and allele distribution in a population of variable length linear structures and presented theoretical results describing the asymptotic behaviour for a GP/GA system evolving in a flat fitness landscape. In addition, we have provided experimental evidence that firmly corroborates the theory, showing a perfect match (within experimental errors) between the predictions of the theory based on generation 0 data and the observed length and allele frequencies at later generations.

The behaviour we have observed and characterised is simple: a) homologous crossovers are totally unbiased with respect to program length, and b) crossover shuffles the alleles present in different individuals and pushes the string distribution towards locus-wise independence.

A mixing behaviour is present in most crossover operators described in the literature on fixed length GAs. It is well known that this destroys "linkage", i.e. correlations, between different allele positions in the population. In the fixed length case the asymptotic convergence towards independence described by Geiringer's theorem is the result of the decay of correlations due to the mixing effect of crossover. Because the representation and operators considered in this paper are generalisations of the corresponding fixed-length ones, it is not so surprising to see that linear GP is also moving towards an independent fixed-point string distribution. In other words, allele mixing is the reason why the right hand side of Equation 9 is a product, like the right hand side of Equation 4. We have no reason to believe that the situation would be significantly different in tree-based GP. Because our extension of Geiringer's theorem is the result of specialising and studying the GP schema theorem's equations, it is not unlikely that in the future we will be able to provide a Geiringer-like theorem for tree-based GP.

Our theoretical results were obtained for the extreme case of infinite populations and flat fitness landscapes. So, why should these be of any relevance to finite GP/GA populations and realistic landscapes? Firstly, because the biases of homologous crossovers in the absence of selection indicate the precise way in which this type of operators would *naturally* tend to explore the search space. When selection is added, the search bias will be modified by the focusing bias of selection, but, except in cases of very strong selection, many of the features of the search bias shown on a flat landscape will be retained. Secondly, because as shown in our experiments, the results obtained with real (but large) populations match very closely the infinite population theory. For smaller populations, the theory can still be used to give short term indications of the behaviour of the system.

Acknowledgements

CRS would like to thank the University of Birmingham for a visiting Professorship and DGAPA-PAPIIT grant IN100201. RP and CRS would like to thank the Royal Society and the University of Essex for their support. Alden Wright did this work while visiting the University of Birmingham, supported by EPSRC grant GR/R47394.

References

Altenberg, L. (1995) The Schema Theorem and Price's Theorem. In *Foundations of Genetic Algorithms 3*, (Whitley, L. D. & Vose, M. D., eds), pp. 23–49 Morgan Kaufmann, Estes Park, Colorado, USA.

Booker, L. B. (1992) Recombination distributions for genetic algorithms. In *FOGA-92, Foundations of Genetic Algorithms*, Vail, Colorado.

Geiringer, H. (1944) On the probability theory of linkage in Mendelian heredity. *Annals of Mathematical Statistics,* **15** (1), 25–57.

Holland, J. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, USA.

Langdon, W. B. & Poli, R. (2002) *Foundations of Genetic Programming*. Springer.

Langdon, W. B., Soule, T., Poli, R. & Foster, J. A. (1999) The evolution of size and shape. In *Advances in Genetic Programming 3*, (Spector, L., Langdon, W. B., O'Reilly, U.-M. & Angeline, P. J., eds),. MIT Press Cambridge, MA, USA pp. 163–190.

McPhee, N. F. & Miller, J. D. (1995) Accurate replication in genetic programming. In *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, (Eshelman, L., ed.), pp. 303–309 Morgan Kaufmann, Pittsburgh, PA, USA.

McPhee, N. F. & Poli, R. (2001) A schema theory analysis of the evolution of size in genetic programming with linear representations. In *Genetic Programming, Proceedings of EuroGP 2001* LNCS Springer-Verlag, Milan.

McPhee, N. F., Poli, R. & Rowe, J. E. (2001) A schema theory analysis of mutation size biases in genetic programming with linear representations. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC2001* pp. 1078–1085 IEEE Press, COEX, World Trade Center, 159 Samseong-dong, Gangnam-gu, Seoul, Korea.

Nordin, P. (1994) A compiling genetic programming system that directly manipulates the machine code. In *Advances in Genetic Programming*, (Kinnear, Jr., K. E., ed.),. MIT Press pp. 311–331.

Nordin, P. & Banzhaf, W. (1995) Complexity compression and evolution. In *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)*, (Eshelman, L., ed.), pp. 310–317 Morgan Kaufmann, Pittsburgh, PA, USA.

O'Neill, M. & Ryan, C. (2001) Grammatical evolution. *IEEE Transaction on Evolutionary Compuation,* **5** (4).

Poli, R. (2000) Exact schema theorem and effective fitness for GP with one-point crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference*, (Whitley, D., Goldberg, D., Cantu-Paz, E., Spector, L., Parmee, I. & Beyer, H.-G., eds), pp. 469–476 Morgan Kaufmann, Las Vegas.

Poli, R. (2001*a*) General schema theory for genetic programming with subtree-swapping crossover. In *Genetic Programming, Proceedings of EuroGP 2001* LNCS Springer-Verlag, Milan.

Poli, R. (2001*b*) Exact schema theory for genetic programming and variable-length genetic algorithms with one-point crossover. *Genetic Programming and Evolvable Machines,* **2** (2), 123–163.

Poli, R. & Langdon, W. B. (1997) A new schema theory for genetic programming with one-point crossover and point mutation. In *Genetic Programming 1997: Proceedings of the Second Annual Conference*, (Koza, J. R., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M., Iba, H. & Riolo, R. L., eds), pp. 278–285 Morgan Kaufmann, Stanford University, CA, USA.

Poli, R. & Langdon, W. B. (1998*a*) On the search properties of different crossover operators in genetic programming. In *Genetic Programming 1998: Proceedings of the Third Annual Conference*, (Koza, J. R., Banzhaf, W., Chellapilla, K., Deb, K., Dorigo, M., Fogel, D. B., Garzon, M. H., Goldberg, D. E., Iba, H. & Riolo, R., eds), pp. 293–301 Morgan Kaufmann, University of Wisconsin, Madison, Wisconsin, USA.

Poli, R. & Langdon, W. B. (1998*b*) Schema theory for genetic programming with one-point crossover and point mutation. *Evolutionary Computation,* **6** (3), 231–252.

Poli, R. & Logan, B. (1996) The evolutionary computation cookbook: recipes for designing new algorithms. In *Proceedings of the Second Online Workshop on Evolutionary Computation*, Nagoya, Japan.

Poli, R. & McPhee, N. F. (2001*a*) Exact GP schema theory for headless chicken crossover and subtree mutation. In *Proceedings of the 2001 Congress on Evolutionary Computation CEC 2001*, Seoul, Korea.

Poli, R. & McPhee, N. F. (2001*b*) Exact schema theorems for GP with one-point and standard crossover operating on linear structures and their application to the study of the evolution of size. In *Genetic Programming, Proceedings of EuroGP 2001* LNCS Springer-Verlag, Milan.

Poli, R. & McPhee, N. F. (2001*c*) Exact schema theory for GP and variable-length GAs with homologous crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)* Morgan Kaufmann, San Francisco, California, USA.

Poli, R., Rowe, J. E. & McPhee, N. F. (2001) Markov chain models for GP and variable-length GAs with homologous crossover. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)* Morgan Kaufmann, San Francisco, California, USA.

Poli, R., Rowe, J. E., Stephens, C. R. & Wright, A. H. (2002) Allele diffusion in linear genetic programming and variable-length genetic algorithms with subtree crossover. In *Proceedings of EuroGP 2002*.

Soule, T., Foster, J. A. & Dickinson, J. (1996) Code growth in genetic programming. In *Genetic Programming 1996: Proceedings of the First Annual Conference*, (Koza, J. R., Goldberg, D. E., Fogel, D. B. & Riolo, R. L., eds), pp. 215–223 MIT Press, Stanford University, CA, USA.

Spears, W. M. (2000) Limiting distributions for mutation and recombination. In *Proceedings of the Foundations of Genetic Algorithms Workshop (FOGA 6)*, (Spears, W. M. & Martin, W., eds), Charlottesville, VA, USA.

Stephens, C. R. (2001) Some exact results from a coarse grained formulation of genetic dynamics. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2001)*, (Spector, L., Goodman, E. D., Wu, A., Langdon, W. B., Voigt, H.-M., Gen, M., Sen, S., Dorigo, M., Pezeshk, S., Garzon, M. H. & Burke, E., eds), pp. 631–638 Morgan Kaufmann, San Francisco, California, USA.

Stephens, C. R. & Waelbroeck, H. (1997) Effective degrees of freedom in genetic algorithms and the block hypothesis. In *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA97)*, (Bäck, T., ed.), pp. 34–40 Morgan Kaufmann, East Lansing.

Stephens, C. R. & Waelbroeck, H. (1999) Schemata evolution and building blocks. *Evolutionary Computation,* **7** (2), 109–124.

Vose, M. D. & Wright, A. H. (2001) Form invariance and implicit parallelism. *Evolutionary Computation,* **9** (3), 355–370.