
Exact Schema Theorem and Effective Fitness for GP with One-Point Crossover

Riccardo Poli
School of Computer Science
The University of Birmingham
Birmingham, B15 2TT, UK
R.Poli@cs.bham.ac.uk
+44-121-414-3739

Abstract

This paper extends recent results in the GP schema theory by formulating a proper exact schema theorem for GP with one-point crossover. This gives an exact expression for the expected number of instances of a schema at the next generation in terms of macroscopic quantities. This result allows the exact formulation of the notion of effective fitness in GP.

1 INTRODUCTION

Schemata are traditionally used to explain why GAs and more recently GP work [1, 2, 3, 4, 5]. Schemata are similarity templates representing sets of points in the search space. Schema theorems are descriptions of how the number of members of the population belonging to a schema vary over time [6].

The usefulness of schema theorems has been often criticised on the basis that they give only a *lower bound* for the *expected value* of the number of instances of a schema H at the next generation $E[m(H, t + 1)]$. The presence of the expectation operator means that it is not easy to use the theorems to predict the behaviour of a genetic algorithm over multiple generations. Also, since schema theorems provide only lower bounds, some researchers argue that their predictions are not useful even for a single generation ahead.

In very recent work [7] I have presented new theoretical results on GP and GA schemata which overcome some of the weaknesses of the schema theorem. For example, I have extended to GP recent work on GA theory [8, 9] and made the effects and the mechanisms of schema creation explicit. This has allowed an exact formulation (rather than a lower bound) for the expected number of instances of a schema at the next generation.

One of the problems with this exact account is that

$E[m(H, t + 1)]$ is expressed as a function of microscopic quantities (i.e. properties of the individuals in the population, like their selection probability) rather than macroscopic quantities (i.e. properties of schemata, like their fitness or number of instances). Nonetheless, this is a very useful result. For example, I used it to derive an improved version of an earlier GP schema theorem [3, 4] in which some schema creation events are accounted for [7].

This paper extends the work described above by reformulating the exact expression for $E[m(H, t + 1)]$ in terms of macroscopic quantities only. This leads to a proper exact schema theorem for GP with one-point crossover. This result allows the use of some of the theory developed for GAs in [8, 9], including the notion of effective fitness (firstly applied to GP in [10, 11]).

The structure of the paper is as follows. Earlier relevant work on GP and GA schemata is reviewed in Section 2. Then, in Section 3 our recent exact formulation for $E[m(H, t + 1)]$ is summarised. Section 4 presents the main contributions of this paper: the macroscopic exact schema theorem and the exact definition of effective fitness for GP with one-point crossover. Section 5 gives a detailed example on how to use the theory in practice. Some conclusions are drawn in Section 6.

2 BACKGROUND

2.1 GP SCHEMA THEORIES

Several alternative definitions of GP schema have been proposed in the literature (see [6, 4] for more details). All of them define schemata as templates composed of one or multiple trees or fragments of trees. In some definitions [12, 13, 14] schema components are *non-rooted* and, therefore, a schema can be present multiple times within the same program. This leads to considerable mathematical difficulties. In more recent definitions [3, 5] schemata are represented by *rooted* trees or tree fragments, which

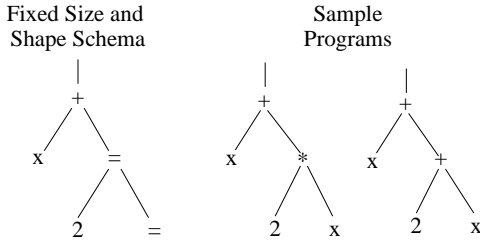


Figure 1: Example of fixed-size-and-shape schema and some instances of programs sampling it.

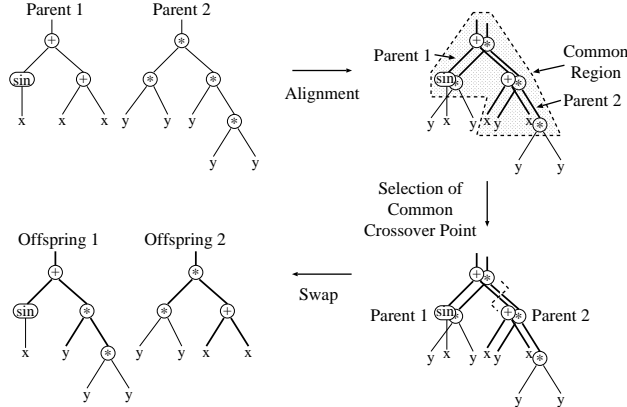


Figure 2: GP one-point crossover. The thick lines are links that can be selected as common crossover points.

make schema theorem calculations easier.

Rosca [5] proposed a definition of schema, called *rooted tree-schema*, in which a schema is a rooted contiguous tree fragment. For example, the rooted tree-schema $(+ \# x)$ represents all the programs whose root node is a $+$ and the second argument of which is x . The symbol $\#$ is “don’t care” symbol which stands for any valid subtree. Rosca derived a schema theorem for GP with standard crossover. One of the problems of this result is that it provides only a lower bound. Another problem is that such a bound is a function of some microscopic quantities.

In [3] we proposed a simpler definition of schema for GP in which a *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 set and the terminal set used in a GP run. The symbol $=$ is a “don’t care” symbol which stands for a *single* terminal or function. A schema H represents programs having the same shape as H and the same labels for the non- $=$ nodes. In the following we will refer to our schemata as *fixed-size-and-shape schemata*. An example of schema with some of its instances is shown in Figure 1 assuming that $\mathcal{F}=\{+, -\}$ and $\mathcal{T}=\{x, y\}$.¹

¹For symmetry, in this paper we use the convention that also the root node of a program or a schema has an output link. This leads to minor changes in the results obtained in our earlier work. Whenever these are reported here we also modify them to take

To derive a GP schema theorem for these schemata a new form of crossover, *one-point crossover*, was used. This works by selecting a common crossover point in the parent programs and then swapping the corresponding subtrees, like standard crossover. To account for the possible structural diversity of the two parents, one-point crossover analyses the two trees from the root nodes and considers for the selection of the crossover point only the parts of the two trees (common region) which have the same topology (i.e. the same arity in the nodes encountered traversing the trees from the root node) [3, 4] as illustrated in Figure 2.

In the absence of mutation, the resulting schema theorem is:

$$E[m(H, t + 1)] \geq Mp(H, t) \left\{ 1 - p_{xo} \left[p_{\text{diff}}(t) (1 - p(G(H), t)) + \frac{\mathcal{L}(H)}{N(H)} (p(G(H), t) - p(H, t)) \right] \right\} \quad (1)$$

where $G(H)$ is the zero-th order schema with the same structure of H where all the defining nodes in H have been replaced with “don’t care” symbols, M is the number of individuals in the population, $p_{\text{diff}}(t)$ is the conditional probability that H is disrupted by crossover when the second parent has a different shape (i.e. does not sample $G(H)$), p_{xo} is the probability of crossover, $E[m(H, t + 1)]$ is the expected number of individuals matching the schema H at generation $t + 1$, $p(H, t)$ is the probability of selection of the schema H , $N(H)$ is the total number of nodes in the schema, and $\mathcal{L}(H)$ is the number of links in the minimum tree fragment including all the non- $=$ symbols within a schema H (see [3, 4] for more details and the proof). In fitness proportionate selection, $p(H, t) = \frac{m(H, t)f(H, t)}{M\bar{f}(t)}$ where $m(H, t)$ is the number of programs matching the schema H at generation t , $f(H, t)$ is the mean fitness of the programs matching H , and $\bar{f}(t)$ is the mean fitness of the programs in the population. The probability $p_{\text{diff}}(t)$ is hard to model mathematically. In the absence of additional information one should assume $p_{\text{diff}}(t) = 1$.

As we noted in [15] the selection/crossover/mutation process can be seen as a Bernoulli trial (a newly created individual either samples or does not sample H) and, therefore, $m(H, t + 1)$ is a binomial stochastic variable. So, if we denote with $\alpha(H, t)$ the success probability of each trial (i.e. the probability that a newly created individual samples H), which we term the *total transmission probability* of H , we have that an exact schema theorem is simply $E[m(H, t + 1)] = M\alpha(H, t)$. Unfortunately, until very recently nobody knew how to express $\alpha(H, t)$ exactly.

this extra link into account.

2.2 EXACT GA SCHEMA THEORY

Thanks to Stephens and Waelbroeck [8, 9] it is now possible to express exactly $\alpha(H, t)$ for GAs operating on fixed-length bit strings. In a GA with one point crossover applied with a probability p_{xo} , $\alpha(H, t)$ is given by:²

$$\alpha(H, t) = (1 - p_{xo})p(H, t) + \frac{p_{xo}}{N-1} \sum_{i=1}^{N-1} p(L(H, i), t)p(R(H, i), t) \quad (2)$$

where $L(H, i)$ is the schema obtained by replacing with “don’t care” symbols (*) all the elements of H from position $i + 1$ to position N , $R(H, i)$ is the schema obtained by replacing with “don’t care” symbols all the elements of H from position 1 to position i , and i varies over the valid crossover points. The symbol L stands for “left part of”, while R stands for “right part of”. For example, if $H = 1*111$, $L(H, 1) = 1****$, $R(H, 1) = **111$, $L(H, 3) = 1*1**$, $R(H, 3) = ***11$.

2.3 EFFECTIVE FITNESS

The concept of effective fitness was introduced in GP in [10, 11] to explain the reasons for bloat and active-code compression. The effective fitness of program j is defined as follows:

$$f_j^e = f_j \left(1 - p_c \frac{C_j^e}{C_j^a} p_j^d \right), \quad (3)$$

assuming fitness proportionate selection. In this equation C_j^a is the number of nodes in program j , C_j^e is the number of nodes in the active part (in contrast to the intron part) of program j , p_c is the crossover probability, p_j^d is the probability that crossover in an active block of program j leads to worse fitness for the offspring of j and f_j is the fitness of individual j . If P_j^t is the proportion of programs j at generation t , P_j^{t+1} is the average proportion of offspring of j which behave like j at generation $t + 1$, and \bar{f}^t is the average population fitness at generation t , then

$$P_j^{t+1} \approx P_j^t \frac{f_j^e}{\bar{f}^t} \quad (4)$$

which describes “the proliferation of individuals from one generation to the next” [11]. The “ \approx ” sign in the equation should really be “ \geq ” but it was used with the justification that the reconstruction of individuals with the same behaviour as j (due to crossover applied to individuals different from j) was a rare event. Equation 4 clearly indicates that an alternative way of interpreting the effects of

²Since Equation 2 was developed using the simpler approach described in [7], it is considerably different from the equivalent results in [8, 9]. However, by performing some minor calculations the GA schema theorem in [8, 9] can be obtained from it.

crossover is to imagine a GA in which selection only is used, but in which each individual is given a fitness f_j^e rather than the original fitness f_j .

Stephens and Waelbroeck [8, 9] independently rediscovered the notion of effective fitness. Using our own notation, as in Section 2.2, the *effective fitness of a schema* is implicitly defined through the equation

$$E \left[\frac{m(H, t+1)}{M} \right] = \frac{m(H, t)}{M} \cdot \frac{f_{\text{eff}}(H, t)}{f(t)},$$

assuming that fitness proportionate selection is used. This has basically the same form as Equation 4. Indeed the two equations represent nearly the same idea, although in different domains. Since $E \left[\frac{m(H, t+1)}{M} \right] = \alpha(H, t)$ and $\frac{m(H, t)}{M f(t)} = \frac{p(H, t)}{f(H, t)}$, one obtains

$$f_{\text{eff}}(H, t) = \frac{\alpha(H, t)}{p(H, t)} f(H, t) \quad (5)$$

$$= f(H, t) \left[1 - p_{xo} \left(1 - \sum_i \frac{p(L(H, i), t)p(R(H, i), t)}{(N-1)p(H, t)} \right) \right]$$

where we used the value of $\alpha(H, t)$ in Equation 2.

Equation 5 is similar to Equation 3, but there are important differences: f_j^e is an *approximation* (of unknown accuracy, being in fact a lower bound) of the true effective fitness of an *individual* in a standard GP system, while $f_{\text{eff}}(H, t)$ is the *true* effective fitness for a *schema* in a standard binary GA. In addition, the true effective fitness $f_{\text{eff}}(H, t)$ of a schema can be bigger than $f(H, t)$ if the building blocks for H are abundant and relatively fit. On the contrary the estimate/bound given by f_j^e is always smaller than f_j (which may be incorrect).

3 GP HYPERSHEMA THEORY

In [16, 7] I extend our fixed-size-and-shape schema theory for GP with one-point crossover obtaining results similar to those in Section 2.2. This section summarises these results.

If one had a population of programs all having exactly the same size and shape, it would be possible to express the total transmission probability of a fixed-size-and-shape schema, in the presence of one-point crossover, in exactly the same way as in Equation 2, i.e.

$$\alpha(H, t) = (1 - p_{xo})p(H, t) + \frac{p_{xo}}{N(H)} \sum_{i=0}^{N(H)-1} p(l(H, i), t)p(u(H, i), t) \quad (6)$$

where: $N(H)$ is the number nodes in the schema H (which is assumed to have the same size and shape of the programs in the population); $l(H, i)$ is the schema obtained

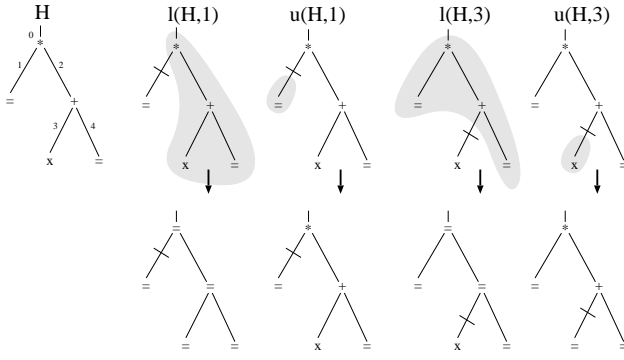


Figure 3: Example of schema and some of its potential fixed-size-and-shape building blocks.

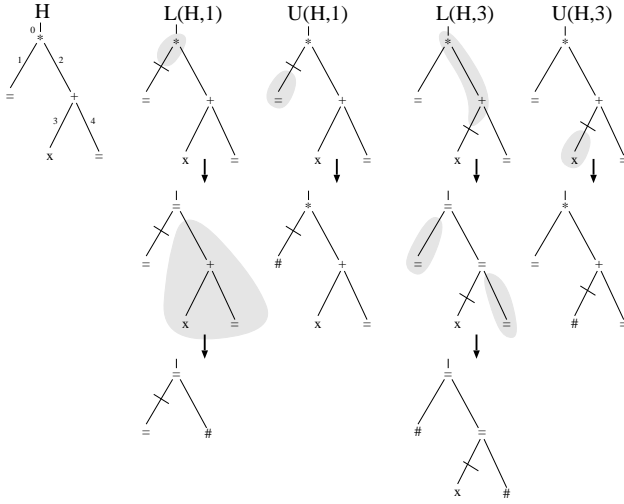


Figure 4: Example of schema and some of its potential hyperschema building blocks.

by replacing all the nodes above crossover point i with $=$ nodes; $u(H, i)$ is the schema obtained by replacing with a $=$ nodes all the nodes below crossover point i ; i varies over the valid $N(H)$ crossover points.³ The proof of this result is reported in [7]. The symbol l stands for “lower part of”, while u stands for “upper part of”. For example, Figure 3 shows how $l(H, 1)$, $u(H, 1)$, $l(H, 3)$, and $u(H, 3)$ are obtained assuming that $H = (* = (+ x =))$ and the crossover points are numbered as in Figure 3 (top left).

To extend this result to populations including programs of different sizes and shapes, in [7] I introduced a more general definition of schema:

Definition 1 GP hyperschema. A *GP hyperschema* is a rooted tree composed of functions from the set $\mathcal{F} \cup \{=\}$ and terminals from the set $\mathcal{T} \cup \{=#\}$. \mathcal{F} and \mathcal{T} are the function set and the terminal set used in a GP run. The operator $=$ is a “don’t care” symbols which stands for exactly

³A simpler version of this equation, valid when $p_{xo} = 1$, was presented in [16] where I failed to state that the equation is only applicable to populations of programs of fixed size and shape.

one node, while the operator $\#$ stands for any valid subtree.

For example the hyperschema is $(* \# (+ x =))$ represents all the programs with the following characteristics: a) the root node is a product, b) the first argument of the root node is any valid subtree, c) the second argument of the root node is $+$, d) the first argument of the $+$ is the variable x , e) the second argument of the $+$ is any valid node in the terminal set. Hyperschemata are a generalisation of both Rosca’s and our fixed-shape-and-size schemata.

Using hyperschemata, I obtained the following general result which is valid for populations of programs of any size and shape (see [7] for the proof):

Theorem 2 Individual-centred Exact GP Schema Thm.

The total transmission probability for a fixed-size-and-shape GP schema H under one-point crossover and no mutation is

$$\alpha(H, t) = (1 - p_{xo})p(H, t) + p_{xo} \sum_{h_1} \sum_{h_2} \frac{p(h_1, t)p(h_2, t)}{NC(h_1, h_2)} \cdot \sum_{i \in C(h_1, h_2)} \delta(h_1 \in L(H, i))\delta(h_2 \in U(H, i)) \quad (7)$$

where: the first two summations are over all the individuals in the population; $NC(h_1, h_2)$ is the number of nodes in the tree fragment representing the common region between program h_1 and program h_2 ; $C(h_1, h_2)$ is the set of indices of the crossover points in such a common region; $\delta(x)$ is a function which returns 1 if x is true, 0 otherwise; $L(H, i)$ is the hyperschema obtained by replacing with $=$ nodes all the nodes on the path between crossover point i and the root node, and with $\#$ nodes all the subtrees connected to the nodes replaced with $=$; $U(H, i)$ is the hyperschema obtained by replacing with a $\#$ node the subtree below crossover point i . If a crossover point i in the common region between two programs is outside the schema H , then $L(H, i)$ and $U(H, i)$ are empty sets.

The hyperschemata $L(H, i)$ and $U(H, i)$ are generalisations of the schemata $l(H, i)$ and $u(H, i)$ used in Equation 6 (compare Figures 3 and 4). They are very important: if one crosses over at point i any individual in $L(H, i)$ and with any individual in $U(H, i)$, the resulting offspring is always an instance of H . Let us try to understand how $L(H, i)$ and $U(H, i)$ are built with an example. If $H = (* = (+ x =))$, as indicated in the second column of Figure 4, $L(H, 1)$ is obtained by first replacing the root node with a $=$ symbol and then replacing the subtree connected to the right of root node with a $\#$ symbol obtaining $(= = \#)$. The schema $U(H, 1)$ is instead obtained by replacing the subtree below the crossover point with a $\#$ symbol obtaining $(* \# (+ x =))$, as illustrated in the third column of Figure 4. The fourth and fifth columns

of Figure 4 show how $L(H, 3) = (= \# (= \times \#))$ and $U(H, 3) = (* = (+ \# =))$ are obtained.

If one restricts the first two summations in Equation 7 to include only the individuals which belong to $G(H)$, one obtains [7]:

Theorem 3 GP S.T. with Schema Creation Correction.

For a fixed-size-and-shape GP schema H under one-point crossover and no mutation

$$\alpha(H, t) \geq (1 - p_{xo})p(H, t) + \frac{p_{xo}}{N(H)} \quad (8)$$

$$\cdot \sum_{i=0}^{N(H)-1} p(L(H, i) \cap G(H), t) p(U(H, i) \cap G(H), t),$$

the equality applying when all the programs in the population sample $G(H)$.

A comparison between the r.h.s. of Equation 1 (divided by M) and the r.h.s. of Equation 8 when $p_m = 0$ and $p_{diff}(t) = 1$ revealed that Theorem 3 provides a better estimate of the true transmission probability of a schema [7]. This is because, unlike the older version, the theorem accounts for some schema creation events.

4 MACROSCOPIC EXACT GP SCHEMA THEOREM AND EFFECTIVE FITNESS

In order to transform Equation 7 into a macroscopic description of schema propagation, let us start by numbering all the possible program shapes, i.e. all the possible fixed-size-and-shape schemata of order 0. Let us denote such schemata as G_1, G_2, \dots . These schemata represent disjoint sets of programs. Their union represents the whole search space. For these reasons, we can rewrite

$$\begin{aligned} \delta(h_1 \in L(H, i)) &= \sum_j \delta(h_1 \in L(H, i) \cap G_j) \\ &= \sum_j \delta(h_1 \in L(H, i)) \delta(h_1 \in G_j). \end{aligned}$$

We substitute this expression and an analogous expression for $\delta(h_2 \in U(H, i))$ into the triple summation in Equation 7 and reorder the terms obtaining:

$$\begin{aligned} &\sum_{h_1} \sum_{h_2} \frac{p(h_1, t) p(h_2, t)}{NC(h_1, h_2)} \sum_{i \in C(h_1, h_2)} \\ &\cdot \sum_j \delta(h_1 \in L(H, i)) \delta(h_1 \in G_j) \\ &\cdot \sum_k \delta(h_2 \in U(H, i)) \delta(h_2 \in G_k) \\ &= \sum_j \sum_k \sum_{h_1} \sum_{h_2} \frac{p(h_1, t) p(h_2, t)}{NC(h_1, h_2)} \end{aligned}$$

$$\begin{aligned} &\cdot \sum_{i \in C(h_1, h_2)} \delta(h_1 \in L(H, i)) \delta(h_1 \in G_j) \\ &\cdot \delta(h_2 \in U(H, i)) \delta(h_2 \in G_k) \\ &= \sum_j \sum_k \sum_{h_1 \in G_j} \sum_{h_2 \in G_k} \frac{p(h_1, t) p(h_2, t)}{NC(h_1, h_2)} \\ &\cdot \sum_{i \in C(h_1, h_2)} \delta(h_1 \in L(H, i)) \delta(h_2 \in U(H, i)) \\ &= \sum_j \sum_k \sum_{h_1 \in G_j} \sum_{h_2 \in G_k} \frac{p(h_1, t) p(h_2, t)}{NC(G_j, G_k)} \\ &\cdot \sum_{i \in C(G_j, G_k)} \delta(h_1 \in L(H, i)) \delta(h_2 \in U(H, i)) \\ &= \sum_j \sum_k \frac{1}{NC(G_j, G_k)} \\ &\cdot \sum_{i \in C(G_j, G_k)} \sum_{h_1 \in G_j} p(h_1, t) \delta(h_1 \in L(H, i)) \\ &\cdot \sum_{h_2 \in G_k} p(h_2, t) \delta(h_2 \in U(H, i)). \end{aligned}$$

From this one obtains the following

Theorem 4 Exact GP Schema Theorem. The total transmission probability for a fixed-size-and-shape GP schema H under one-point crossover and no mutation is

$$\begin{aligned} \alpha(H, t) &= (1 - p_{xo})p(H, t) + p_{xo} \sum_j \sum_k \frac{1}{NC(G_j, G_k)} \\ &\cdot \sum_{i \in C(G_j, G_k)} p(L(H, i) \cap G_j, t) p(U(H, i) \cap G_k, t) \quad (9) \end{aligned}$$

The sets $L(H, i) \cap G_j$ and $U(H, i) \cap G_k$ either are (or can be represented by) fixed-size-and-shape schemata or are the empty set \emptyset . So, the theorem indeed expresses the total transmission probability of H only using the selection probabilities of a set of lower order schemata.

This theorem is a generalisation of Equation 8 which can be obtained from Equation 9 by considering only one term in the summations in j and k (the term for which $G_j = G_k = G(H)$). The theorem is also a generalisation of Equation 6.

Once the value of $\alpha(H, t)$ is available, it is easy to extend to GP with one-point crossover the notion of effective fitness provided in [8, 9]. By using the definition in Section 2.3 and the value of $\alpha(H, t)$ in Equation 9, we obtain:

$$\begin{aligned} f_{\text{eff}}(H, t) &= \frac{\alpha(H, t)}{p(H, t)} f(H, t) \quad (10) \\ &= f(H, t) \left[1 - p_{xo} \left(1 - \sum_j \sum_k \sum_{i \in C(G_j, G_k)} \frac{p(L(H, i) \cap G_j, t) p(U(H, i) \cap G_k, t)}{NC(G_j, G_k) p(H, t)} \right) \right]. \end{aligned}$$

This equation gives the *true effective fitness for a GP schema* under one-point crossover: it is not an approximation or a lower bound. Thanks to this definition it is easy to see that the effective fitness of a GP schema can be bigger than its actual fitness if its building blocks are abundant and relatively fit. This shows that crossover does not always have the destructive connotation often attributed to it in the GP literature (e.g. [13, 11]).

5 EXAMPLE

Since the calculations involved in applying the exact GP schema theorems may become quite lengthy, we will consider one of the simplest non-trivial examples possible.

Let us imagine that we have a function set $\{A_f, B_f, C_f, D_f, E_f\}$ including only 1-arity functions, and the terminal set $\{A_t, B_t, C_t, D_t, E_t\}$. So, for example, a program in this search space might look like $(A_f(B_f B_t))$. Since, the arity of all functions is 1, we can remove the parentheses from the expression obtaining $A_f B_f B_t$. In addition, since the only terminal in each tree is the rightmost node, we can remove the subscripts without generating any ambiguity, obtaining ABB . This can be done for every member of the search space, which can be seen as the space of variable-length strings over the alphabet $\{A, B, C, D, E\}$. So, in this example GP with one-point crossover is really a non-binary variable-length GA.

Let us now consider the schema $AB=$. We want to measure its total transmission probability under fitness proportionate selection and one-point crossover (with $p_{xo} = 1$) in two slightly different populations:

Population 1	Fitness	Population 2	Fitness
AB	2	AB	2
BCD	2	BCD	2
ABC	4	ABC	4
ABCD	6	BCDE	6

In order to do that we need first to compute the “lower part” and “upper part” building blocks of $AB=$. These are:

i	$L(AB=, i)$	$U(AB=, i)$
0	AB=	#
1	=B=	A #
2	===	AB#
3	\emptyset	\emptyset
\vdots	\vdots	\vdots

Let us start by calculating $\alpha(AB=, t)$ for Population 1, using Equation 7:

$$\begin{aligned}
\alpha(AB=, t) &= \sum_{h_1, h_2} \frac{p(h_1, t)p(h_2, t)}{NC(h_1, h_2)} \cdot \sum_i \delta(h_1 \in L(AB=, i))\delta(h_2 \in U(AB=, i)) \\
&= \underbrace{\frac{2}{14} \times \frac{2}{14} \times \frac{1}{2}}_{h_1=AB, h_2=AB, i=0,1} \times (0 \times 1 + 0 \times 1) \\
&+ \underbrace{\frac{2}{14} \times \frac{2}{14} \times \frac{1}{2}}_{h_1=AB, h_2=BCD, i=0,1} \times (0 \times 1 + 0 \times 0) + \dots \quad (13 \text{ terms are omitted}) \\
&+ \underbrace{\frac{6}{14} \times \frac{6}{14} \times \frac{1}{4}}_{h_1=ABCD, h_2=ABCD, i=0,1,2,3} \times (1 \times 1 + 1 \times 1 + 1 \times 1 + 1 \times 1) \\
&= \frac{43}{147} \approx 0.2925
\end{aligned}$$

So, clearly this is a lengthy calculation, which can only produce a numerical result. It cannot really be used to understand how instances of $AB=$ are created in different populations.

Let us now use Equation 9 to do the same calculation. First we need to number all the possible program shapes G_1, G_2 , etc.. Let G_1 be =, G_2 be ==, G_3 be === and G_4 be ====. We do not need to consider other, bigger shapes because the population does not contain any larger programs (i.e. $G_l = \emptyset$ for $l > 4$). Then we need to identify the schemata resulting from calculating $L(AB=, i) \cap G_j$ for all meaningful values of i and j :

		$L(AB=, i) \cap G_j$			
		j			
i		1	2	3	4
0	\emptyset	\emptyset	\emptyset	AB=	\emptyset
1	\emptyset	\emptyset	\emptyset	=B=	\emptyset
2	\emptyset	\emptyset	\emptyset	===	\emptyset
3	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

We do the same for $U(AB=, i) \cap G_k$, obtaining:

		$U(AB=, i) \cap G_k$			
		k			
i		1	2	3	4
0	=	=	==	===	====
1	\emptyset	A=	A=	A==	A===
2	\emptyset	\emptyset	\emptyset	AB=	AB==
3	\emptyset	\emptyset	\emptyset	\emptyset	\emptyset

Finally we need to evaluate the shape of the common regions to determine $NC(G_j, G_k)$ and the links in $C(G_j, G_k)$ for all valid values of j and k . In general this

can be naturally represented using the program shapes G_1 , G_2 , etc.. For the example under consideration:

		Shape of Common Region			
		k			
j	1	2	3	4	
1	G_1	G_1	G_1	G_1	
2	G_1	G_2	G_2	G_2	
3	G_1	G_2	G_3	G_3	
4	G_1	G_2	G_3	G_4	

By using this and the previous tables we can simplify Equation 9 removing all the null terms as follows:

$$\begin{aligned}
\alpha(\text{AB}=\, , t) &= \sum_{j,k} \frac{1}{NC(G_j, G_k)} \cdot \sum_i p(L(\text{AB}=\, , i) \cap G_j, t) p(U(\text{AB}=\, , i) \cap G_k, t) \\
&= \underbrace{\frac{1}{1} \times \sum_{j=1, k=1, i=0} p(L(\text{AB}=\, , i) \cap G_j, t) p(U(\text{AB}=\, , i) \cap G_k, t)}_{\text{6 more terms}} \\
&+ \frac{1}{1} \times [0 \times P(==)] + \dots \quad (\text{6 more terms}) \\
&+ \frac{1}{1} \times [P(AB =)P(==)] \\
&+ \frac{1}{2} \times [P(AB =)P(==) + P(= B =)P(A =)] \\
&+ \dots \quad (\text{6 more terms})
\end{aligned}$$

where for brevity we have used the notation $P(\cdot)$ to represent $p(\cdot, t)$. Simplifying yields

$$\begin{aligned}
\alpha(\text{AB}=\, , t) &= P(AB =)P(=) + \frac{1}{2}P(AB =)P(==) \\
&+ \frac{1}{2}P(= B =)P(A =) + \frac{1}{3}P(AB =)P(===) \\
&+ \frac{1}{3}P(= B =)P(A ==) + \frac{1}{3}P(====)P(AB =) \\
&+ \frac{1}{3}P(AB =)P(=====) + \frac{1}{3}P(= B =)P(A ===) \\
&+ \frac{1}{3}P(====)P(AB ==)
\end{aligned}$$

The complexity of this equation can be reduced by using hyperschemata to represent groups of schemata, obtaining:

$$\begin{aligned}
\alpha(\text{AB}=\, , t) &= P(AB =)P(=) + \frac{1}{2}P(AB =)P(==) \\
&+ \frac{1}{2}P(= B =)P(A =) + \frac{1}{3}P(AB =)P(== \#) \\
&+ \frac{1}{3}P(= B =)P(A = \#) + \frac{1}{3}P(====)P(AB \#)
\end{aligned}$$

This is equivalent to reordering the terms by size and shape of common region and then by crossover point.

This equation is quite different from the one obtained with the earlier exact schema theorem. It is general, i.e. independent from a particular population. Also, it clearly indicates how individuals sampling AB= can be assembled from individuals having different shapes and nodes. The schemata in this equation are the real *building blocks* for AB=.

If we calculate the probabilities of selection of the schemata in the previous equation using Population 1, we obtain:

$$\begin{aligned}
\alpha(\text{AB}=\, , t) &= \frac{4}{14} \times 0 + \frac{1}{2} \times \frac{4}{14} \times \frac{2}{14} + \frac{1}{2} \times \frac{4}{14} \times \frac{2}{14} \\
&+ \frac{1}{3} \times \frac{4}{14} \times \frac{12}{14} + \frac{1}{3} \times \frac{4}{14} \times \frac{10}{14} + \frac{1}{3} \times \frac{6}{14} \times \frac{10}{14} \\
&= \frac{43}{147} \approx 0.2925
\end{aligned}$$

The result is the same in the two calculations as expected. However, once the exact macroscopic formulation of the transmission probability of a schema is available, this is much easier to use in calculations than the corresponding microscopic description. Indeed, we can use it to calculate $\alpha(\text{AB}=\, , t)$ for Population 2 with a simple pocket calculator obtaining $\alpha(\text{AB}=\, , t) \approx 0.1905$. For comparison, for either population, the schema theorem with schema creation correction would have provided the lower bound:

$$\begin{aligned}
\alpha(\text{AB}=\, , t) &\geq \frac{1}{3}P(AB =)P(==) \\
&+ \frac{1}{3}P(= B =)P(A ==) + \frac{1}{3}P(====)P(AB =) \\
&\approx 0.1088
\end{aligned}$$

which is nearly one third of the correct value for Population 1 and a half of the correct value for Population 2. This is because the theorem accounts only for schema creation events in $G(\text{AB}=\,)$. Since there are no creation events of this type in this example, the old GP schema theorem (Equation 1) gives exactly the same bound.

With these results we can now compute the effective fitness for the schema AB= for both populations. For Population 1

$$f_{\text{eff}}(\text{AB}=\, , t) = \frac{\alpha(\text{AB}=\, , t)}{p(\text{AB}=\, , t)} f(\text{AB}=\, , t) \approx 4.1,$$

i.e. thanks to the collaboration of other schemata the schema AB= propagates faster with 100% crossover than with no crossover at all! On the contrary for Population 2

$$f_{\text{eff}}(\text{AB}=\, , t) = \frac{\alpha(\text{AB}=\, , t)}{p(\text{AB}=\, , t)} f(\text{AB}=\, , t) \approx 2.7.$$

So, in Population 2 the schema is effectively a below-average schema ($f_{\text{eff}}(\text{AB}=\, , t) = 2.7 < f(t) = 3.5$) despite the fact that its fitness ($f(\text{AB}=\, , t) = 4$) is above average.

6 CONCLUSIONS

In this paper, a macroscopic exact schema theorem for genetic programming with one-point crossover is provided. This theorem extends to GP (and also to a form of variable length GAs) recent GA theory, using a more general notion of GP schema: the hyperschema.

Thanks to this theorem it is now possible to express exactly the notion of effective fitness. This was originally introduced in GP in approximate form in [10, 11] to explain the reasons for bloat and active-code compression, and later re-defined more formally for GAs in [8, 9] to describe the effects of the search operators on the reproductive efficiency of a schema. So, this paper establishes a formal link between the two main areas of theoretical research in GP: the study of bloat and the theory of schemata. In future research I hope to be able to use this rigorous formulation of the effective fitness to model mathematically and understand better the reasons for bloat, intron proliferation and code compression.

ACKNOWLEDGEMENTS

The author would like to thank the members of the EE-BIC (Evolutionary and Emergent Behaviour Intelligence and Computation) group at Birmingham and W. B. Langdon for useful discussions and comments, and Nic McPhee for his exceptionally thorough and helpful review.

References

- [1] J. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, Massachusetts: MIT Press, second ed., 1992.
- [2] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, Massachusetts: Addison-Wesley, 1989.
- [3] R. Poli and W. B. Langdon, "A new schema theory for genetic programming with one-point crossover and point mutation," in *Genetic Programming 1997: Proceedings of the Second Annual Conference* (J. R. Koza, K. Deb, M. Dorigo, D. B. Fogel, M. Garzon, H. Iba, and R. L. Riolo, eds.), (Stanford University, CA, USA), pp. 278–285, Morgan Kaufmann, 13-16 July 1997.
- [4] R. Poli and W. B. Langdon, "Schema theory for genetic programming with one-point crossover and point mutation," *Evolutionary Computation*, vol. 6, no. 3, pp. 231–252, 1998.
- [5] J. P. Rosca, "Analysis of complexity drift in genetic programming," in *Genetic Programming 1997: Proceedings of the Second Annual Conference* (J. R. Koza, K. Deb, M. Dorigo, D. B. Fogel, M. Garzon, H. Iba, and R. L. Riolo, eds.), (Stanford University, CA, USA), pp. 286–294, Morgan Kaufmann, 13-16 July 1997.
- [6] R. Poli and W. B. Langdon, "A review of theoretical and experimental results on schemata in genetic programming," in *Proceedings of the First European Workshop on Genetic Programming* (W. Banzhaf, R. Poli, M. Schoenauer, and T. C. Fogarty, eds.), vol. 1391 of *LNCS*, (Paris), pp. 1–15, Springer-Verlag, 14-15 Apr. 1998.
- [7] R. Poli, "Hyperschema theory for GP with one-point crossover, building blocks, and some new results in GA theory," in *Genetic Programming, Proceedings of EuroGP 2000* (R. Poli, W. Banzhaf, and *et al.*, eds.), Springer-Verlag, 15-16 Apr. 2000.
- [8] C. R. Stephens and H. Waelbroeck, "Effective degrees of freedom in genetic algorithms and the block hypothesis," in *Proceedings of the Seventh International Conference on Genetic Algorithms (ICGA97)* (T. Bäck, ed.), (East Lansing), Morgan Kaufmann, 1997.
- [9] C. R. Stephens and H. Waelbroeck, "Schemata evolution and building blocks," *Evolutionary Computation*, vol. 7, no. 2, pp. 109–124, 1999.
- [10] P. Nordin and W. Banzhaf, "Complexity compression and evolution," in *Genetic Algorithms: Proceedings of the Sixth International Conference (ICGA95)* (L. Eshelman, ed.), (Pittsburgh, PA, USA), pp. 310–317, Morgan Kaufmann, 15-19 July 1995.
- [11] P. Nordin, F. Francone, and W. Banzhaf, "Explicitly defined introns and destructive crossover in genetic programming," in *Proceedings of the Workshop on Genetic Programming: From Theory to Real-World Applications* (J. P. Rosca, ed.), (Tahoe City, California, USA), pp. 6–22, 9 July 1995.
- [12] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.
- [13] U.-M. O'Reilly and F. Oppacher, "The troubling aspects of a building block hypothesis for genetic programming," in *Foundations of Genetic Algorithms 3* (L. D. Whitley and M. D. Vose, eds.), (Estes Park, Colorado, USA), pp. 73–88, Morgan Kaufmann, 31 July–2 Aug. 1994 1995.
- [14] P. A. Whigham, "A schema theorem for context-free grammars," in *1995 IEEE Conference on Evolutionary Computation*, vol. 1, (Perth, Australia), pp. 178–181, IEEE Press, 29 Nov. - 1 Dec. 1995.
- [15] R. Poli, W. B. Langdon, and U.-M. O'Reilly, "Analysis of schema variance and short term extinction likelihoods," in *Genetic Programming 1998: Proceedings of the Third Annual Conference* (J. R. Koza, W. Banzhaf, K. Chellapilla, K. Deb, M. Dorigo, D. B. Fogel, M. H. Garzon, D. E. Goldberg, H. Iba, and R. Riolo, eds.), (University of Wisconsin, Madison, Wisconsin, USA), pp. 284–292, Morgan Kaufmann, 22-25 July 1998.
- [16] R. Poli, "Schema theory without expectations for GP and GAs with one-point crossover in the presence of schema creation," in *Foundations of Genetic Programming* (T. Haynes, W. B. Langdon, U.-M. O'Reilly, R. Poli, and J. Rosca, eds.), (Orlando, Florida, USA), 13 July 1999.