Schema Disruption in Chromosomes That Are Structured as Binary Trees

William A. Greene

Computer Science Department
University of New Orleans
New Orleans, LA 70148
bill@cs.uno.edu

Abstract. We are interested in schema disruption behavior when chromosomes are structured as binary trees. We give the definition of the disruption probability $dp(H)$ of a schema H, and also the relative diameter $rel\Delta(H)$ of H. We show that in the general case that $dp(H)$ can far exceed $rel\Delta(H)$, but when the chromosome is a complete binary tree then the inequality $dp(H) \leq rel\Delta(H)$ holds almost always. Thus the more compactly the tree chromosome is structured, the better is the behavior to be expected from geneticism.

1 Introduction

The field of Genetic Algorithms (GAs) is a heuristic problem-solving paradigm which is inspired by the machinations of evolution. There is some problem of interest at hand. There are candidate solutions to the problem, some of which are better (or fitter) than others. Usually the number of possible candidate solutions is enormous, too large to search exhaustively. In GAs, a population of candidate solutions is maintained; this population is a small sampling of the full solution space. The population is subjected to such evolutionary forces as survival of the fittest, mating with crossover, and mutation. The hope is that as the population evolves, fitter and fitter solutions will appear.

Candidate solutions differ one from another by having different property values for certain properties that are pertinent to the problem at hand. We represent (identify) the candidate solution with the property values that characterize it.

In classical Genetic Algorithms as invented by Holland [1] and popularized by Goldberg [2], candidate solutions (which we will begin to refer to as *individuals* and *chromosomes*) in the population are bits strings, all of the same length N. Thus, each is an element of the Cartesian product space $\{0, 1\}^N$. We here emphasize that the bits are arranged in a linear sequence. For a theoretical analysis of the convergence behavior of an evolving population, the notion of the *schema* is introduced. Let the symbol * be a don't-care symbol. Then a schema H (for hyperplane) is defined to be an element of $\{0, 1, *\}^N$. The positions in H which are the don't-care symbol are termed the *free positions*; the positions occupied by 0's or 1's are the *fixed positions*. A schema stands for an entire subspace of the possible bit strings, namely, those bit strings which agree with the schema at its fixed positions. An element of this subspace is termed a *repre-*

sentative of schema H. The *defining length* $\delta(H)$ of a schema means the distance between its outermost fixed positions. For example, $\delta(\ (*, *, 0, 1, *, *, *, 1, *)\ )$ equals 5.

The term *building block* is used to signify a group of related bits, plus values (0's or 1's) for them, that enhance the fitness of an individual whose bits are so assigned. Clearly, a schema gives the characterizing properties of a building block.

Under *one-point crossover* for mating by two parental chromosomes, we uniformly randomly choose one of the $N - 1$ links between the N bits, cut both parents at that same point, and the parental fragments that result are interchanged to form the two children. A cutpoint is said to *disrupt* a schema if it falls between the two outermost fixed positions. The *disruption probability dp(H)* is the probability that this occurs, and clearly $dp(H)$ equals $\delta(H) / (N - 1)$. The term disruption is appropriate, since if one parent is a representative of H, and the cutpoint falls between the two outermost fixed positions, then it is possible that neither child is again a representative of H. Thus, building blocks can fall away under mating with crossover. Clearly the disruption probability of a schema (or building block) is diminished if the pertinent bits are located close together.

There is an obvious weakness of the standard linear arrangement of bits in a chromosome: a bit has two nearest bits, not more. What if it is in the nature of the problem at hand that a bit should be equally close to three, or four, or more, other bits?

We are interested in chromosomes whose bits are arranged in ways other than as a linear sequence. In particular we are interested in chromosomes structured as binary trees.

Alternative bit arrangements have frequently been used in applications written up in the literature. As for theoretical study, non-linear bit arrangements and a schema theory for such, also have been studied. For instance, Greene [3] has a non-linear schema theorem which may apply when the chromosome is structured as an arbitrary connected graph. As for chromosomes structured as trees, study of schema theory for them has principally come from those in the Genetic Programming (GP) community. In GP approaches, individuals are programs, specifically functions, realized as expression trees. Mating with crossover consists of clipping out and exchanging subtrees between the two parents. The individuals in a population can have quite different shapes, which fact complicates a number of issues, such as, what will be the definition of a schema, and what relation will hold between the locations of the cutpoints in the two parents? For Koza [4], O'Reilly [5], O'Reilly & Oppacher [6], and Whigham [7], schemas are expression fragments which incorporate don't-care symbols, and which are further characterized by not being anchored to some fixed position within the expression tree and moreover can be instantiated multiple times within the same individual. In Rosca [8], the innovation is that a schema is an expression fragment which is anchored at the root of the expression tree.

Our own interest in non-linear bit arrangements did not originate from a prior interest in genetic programming. Rather, our intuition has been that strictly linear bit arrangements are simply too confining and too inflexible for GAs. Furthermore, we envision a population of chromosomes that all have the same shape. From within the

GP community, the work that comes closest to our own efforts is that of Poli & Langdon [9]. Their definitions of schema, mutation, and crossover are the closest carryover to GP of the allied notions from standard GAs with linear bit arrangements. For Poli & Langdon, a schema is a rooted tree of symbols, where the root is to correspond to the root of an expression tree that is an individual in the population. Below, we will remark on similarities between our present research and the approach of Poli & Langdon.

In this paper we are interested specifically in the disruption probability $dp(H)$ of schemas H in chromosomes structured as binary trees. Knowing about the value of $dp(H)$ must figure in the statement and proof of any schema theorem akin to the classic one by Holland (confer [1] or [2]). A closed expression that exactly calculates $dp(H)$ for arbitrary H is likely hard to come up with. Hence we seek an expression which is more easily calculated and which may be an upper bound for $dp(H)$. An insight which should be carried over from the linear case is that we should explore how $dp(H)$ is related to how closely situated together are the fixed positions of schema H.

2   Binary Trees

We assume the reader is familiar with binary trees. Trees consist of *nodes*, connected by *edges*. All the binary trees we consider are finite. We use T to name a binary tree. The *level* of a node: the root of the tree is at level zero; the level of a non-root node is one greater than the level of its parent. A *level is full* if it contains the maximum possible number of nodes, which is 2^{level}. A binary *tree is full* if its every level is full. A binary *tree is complete* if its every level is full, except possibly the bottom level. (The usual definition of complete insists that the leaves on the bottom level are packed together without gaps off to the left, but we do not need this stipulation.)

There is a unique path between any two nodes in a binary tree. We define the distance between two tree nodes to be the length of that path. This notion of distance satisfies the triangle inequality, $dist(x, z) \leq dist(x, y) + dist(y, z)$. Given a set S of nodes in the tree, the diameter $\Delta(S)$ is the maximum distance between any two elements of S.

An individual whose bits are linked as the nodes in some binary tree is an easily comprehended concept. A *schema* will be the obvious analogue from linear chromosomes: imagine replacing some of the bits (0's or 1's) with the don't-care symbol. We define the *relative diameter* $rel\Delta(H)$ of a schema H to be the ratio $\Delta(fixed(H)) / \Delta(T)$, where $fixed(H)$ is the set of fixed positions of H. We abbreviate $\Delta(fixed(H))$ to simply $\Delta(H)$. Moreover, we sometimes will use the same name H to designate just the fixed nodes of schema H. The relative diameter $rel\Delta(H)$ captures the notion of how closely together the fixed positions of H are situated in the host chromosome T.

We assume all our individuals have the same shape as binary trees. Cutting one edge in a tree divides the tree into two connected subtrees; these fragments are to be used to construct two children at crossover time.

Cutting a tree edge *separates two nodes x and y* if that edge is one lying in the unique path between x and y; in this case x and y end up in different fragmental sub-

trees. A cut *separates a set S* of tree nodes if there are two nodes in *S* which are separated by the cut. A cut *disrupts* a schema *H* if it separates the fixed positions of H. We intend to practice mating with crossover by uniformly randomly cutting one edge of chromosomal tree T, and interchanging parental fragments to form the children. Hence we will define the *disruption probability dp(H)* of schema *H* to be the fraction (number of edges that disrupt *H*) / (total number of edges in *T*).

Given a schema *H* of tree *T*, we let T_H denote the smallest subtree of *T* which contains all the fixed positions of *H*. T_H unequivocally exists, for it is the intersection of all the subtrees of T which contain the fixed positions of H. (Our research was done independently from Poli & Langdon [], but there is an overlap of ideas. The number of edges that disrupt *H* comes closest to what they term the defining length of a tree schema, and they give the name minimum tree fragment to T_H.)

We state without proof the following results.

Proposition 1: (a) Each leaf of T_H is an element of *H*;
   (b) $\Delta(H) = \Delta(T_H)$, and $rel\Delta(H) = rel\Delta(T_H)$;
   (c) The set of *T*-edges that separate *H* is the same as the set that separate T_H;
   (d) $dp(H) = dp(T_H)$.

We will be interested in if and when the relation $dp(H) \leq rel\Delta(H)$ holds. First we observe that a linear sequence of bits (as in classical GAs) is in particular also a binary tree of bits, and in this case $dp(H)$ equals $rel\Delta(H)$ and both equal $\delta(H) / (N - 1)$ where, recall, $\delta(H)$ is the defining length of *H* and *N* is the number of bits in the sequence.

3   The General Case: Arbitrary Binary Trees

Proposition 2: There is no constant $k > 0$ which will make the inequality
$dp(H) \leq k \cdot rel\Delta(H)$ hold for arbitrary schemas *H* in arbitrary binary trees *T*.

   Proof: Consider the binary tree *T* illustrated in Figure 1. The left subtree T_l of root *r* is a full binary tree of height *h*. The (fixed nodes of) schema *H* consists of the nodes on the bottom level of subtree T_l. The rest of tree *T* besides T_l consists of the root *r* and the depicted nodes trailing off from it in a line to the right; we term all these (including *r*) as tail nodes and we let *t* be the number of them. The smallest subtree T_H of *T* which contains (the fixed nodes of) *H* is the left subtree T_l, and it has $2^{h+1} - 1$ nodes and therefore $2^{h+1} - 2$ edges. The tail nodes add another *t* edges to tree *T*, and it follows that $dp(H) = \dfrac{2^{h+1} - 2}{2^{h+1} - 2 + t}$. We will soon choose $t > h$, in which case $rel\Delta(H) = \dfrac{2h}{h+t}$. Now consider the ratio
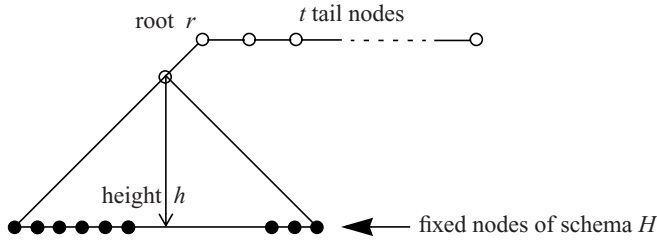
Fig. 1. Example of a wayward binary tree

$$R = \frac{dp(H)}{rel\Delta(H)} = \frac{2^{h+1} - 2}{2^{h+1} - 2 + t} \cdot \frac{h+t}{2h} \ .$$

If we choose $t = 2^{h+1}$, then $\dfrac{2^{h+1} - 2}{2^{h+1} - 2 + t} \approx \dfrac{1}{2}$ and hence ratio $R \approx \dfrac{h+t}{4h}$, which

can be made arbitrarily large. This proposition now follows.

4 Complete Binary Trees

Proposition 3: The inequality $dp(H) \le rel\Delta(H)$ holds for all schemas H and all complete binary trees T, with the exception of certain small trees T, and certain schemas H containing very large numbers of fixed positions.

  Proof: Recall the notation, T_H is the smallest subtree of T which contains (the fixed positions of) H. If 1 is the value of the fraction $dp(H)$ = (number of edges that separate T_H) / (number of edges in T), it follows that $T_H = T$, then $\Delta(H) = \Delta(T_H) = \Delta(T)$, then $rel\Delta(H) = 1$, so that $dp(H) = rel\Delta(H)$. Thus, the interesting case is when the fraction $dp(H)$ is strictly less than 1.

Given a certain diameter Δ, there are many schemas H which have that diameter. Some are large and some are small, and the same can be said for the enclosing subtree T_H of H. Now imagine the diameter value as a given. We will find an upper bound for fraction $dp(H)$, by calculating the most that its numerator can be, and then the least its denominator can be and still exceed the numerator.

The numerator of $dp(H)$ can be as large as the number of edges in the largest possible subtree T_H whose diameter is the given diameter value. Let h_d be a deepest node of H. We introduce some notation: let h be the height of tree T; let d be the depth of h_d; let $\delta = \Delta(H)$. Every node of T_H must be within distance $\delta = \Delta(H)$ of any

given element of H, in particular this is so for h_d. We will count how many nodes can possibly be in our chromosomal tree T, be no deeper than h_d, and be at distance at most δ from h_d. Subtree T_H can be as big as that set of nodes. (For emphasis let us note that according to Proposition 1, also the set of fixed nodes of H can be that big.) After we have determined T_H, we will find the smallest complete binary tree T of height h such that $T \supseteq T_H$. These two, T and T_H, will determine the largest that fraction $dp(H)$ can be, for the given diameter value.

Either diameter δ is even or it is odd. And either $\delta \leq d$ or $\delta > d$. In our proof we will consider four cases. We will give full details for two of the cases and leave the details of the other two cases to the reader.

Case I: even $\delta \leq d$: (See Figure 3 for guidance.) Consider the path of length δ, consisting of ancestor nodes from h_d towards the root of T; denote the nodes on this path as $h_d = v_0, v_1, v_2, ..., v_\delta$. Subtree T_H could contain all the nodes in a full subtree,
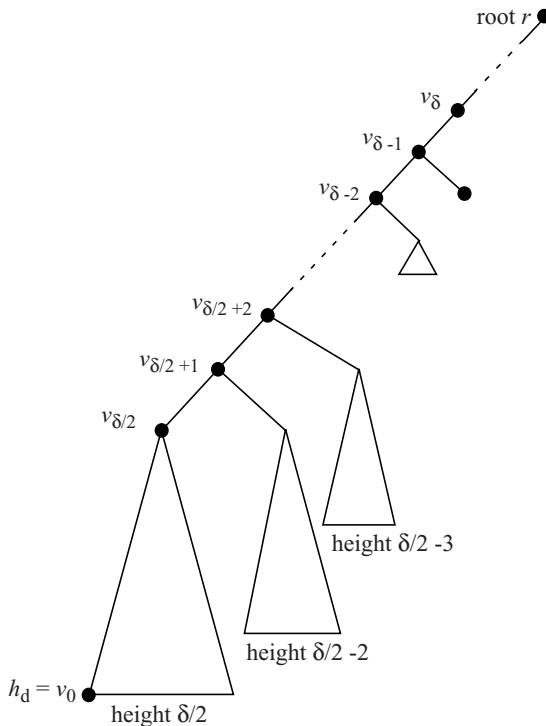

Fig. 2. δ is even, $\delta \leq d$

rooted at $v_{\delta/2}$ and having height $\delta/2$. Similarly, T_H could contain all the nodes in a full subtree, rooted at the second child of $v_{\delta/2+1}$, and having height $\delta/2 - 2$. Similarly for a full subtree, rooted at the second child of $v_{\delta/2+2}$, and having height $\delta/2 - 3$. And so on, back to a full subtree, rooted at the second child of $v_{\delta-2}$, and having height 1; then a full subtree of height 0 that consists exactly of the second child of $v_{\delta-1}$, and, finally, node v_δ by itself, of course. How many nodes have we to count? There are the $\delta/2$ full subtrees we have identified, and also the $\delta/2$ ancestor nodes $v_{\delta/2+1}, \ldots, v_\delta$ which are not inside the identified subtrees. The sizes of the full subtrees are the exceptional size $2^{\delta/2+1} - 1$, then sizes $2^{\delta/2-1} - 1$, $2^{\delta/2-2} - 1$, down to $2^1 - 1$. Altogether the number of nodes we are tallying is $2^{\delta/2+1} + (2^{\delta/2-1} + 2^{\delta/2-2} + \ldots + 2^1) = 2^{\delta/2+1} + (2^{\delta/2} - 2) = 3 \cdot 2^{\delta/2} - 2$. This latter value is the most nodes that T_H can have, hence the most edges that can be in T_H is one less, or $3 \cdot 2^{\delta/2} - 3$.

Continuing with Case I, we now consider host chromosomal tree T. If the depth d of node h_d is less than the height h of T, then to be a complete tree of height h, T can be as small as having only one node on level h, in which case T has 2^h nodes and therefore $2^h - 1$ edges. But if the depth d of h_d equals the height h of T, then since we have allowed T_H to be as big as containing the full subtree of height $\delta/2$ rooted at node $v_{\delta/2}$, it follows that T will be required to have at least $2^{\delta/2}$ nodes on its bottom level h. Then T must have at least $2^h + 2^{\delta/2} - 1$ nodes and hence at least $2^h + 2^{\delta/2} - 2$ edges. Ergo, $dp(H)$ is bounded above by $\dfrac{3 \cdot 2^{\delta/2} - 3}{2^h - 1}$ if $d < h$, but

bounded above by $\dfrac{3 \cdot 2^{\delta/2} - 3}{2^h + 2^{\delta/2} - 2}$, if $d = h$.

Since T is complete but not necessarily full, $\Delta(T)$ is either $2h$ or $2h - 1$; in either event, $rel\Delta(H) \geq \dfrac{\delta}{2h}$.

Putting our two bounds together, and for now focusing on the possibility that $d < h$, our task is to determine if or when $\dfrac{3 \cdot 2^{\delta/2} - 3}{2^h - 1} \leq \dfrac{\delta}{2h}$, or equivalently, if or when

$$\frac{3 \cdot 2^{\delta/2} - 3}{\delta/2} \leq \frac{2^h - 1}{h}$$. Define the function f by $f(n) = \frac{2^n - 1}{n}$. At issue is if or when

$3 \cdot f(\delta/2) \leq f(h)$. Since function f is near-exponential, certainly this latter inequality will hold if δ is not too near its maximum value of $2h$, except the inequality may possibly fail for small values of h. On the other hand, in Case I, $\delta \leq d \leq h$ and so δ is not near $2h$.

We used a computer program to examine if or when the inequalities

$$\frac{3 \cdot 2^{\delta/2} - 3}{2^h - 1} \leq \frac{\delta}{2h} \qquad \text{, for even } \delta \leq d < h \text{, and}$$

$$\frac{3 \cdot 2^{\delta/2} - 3}{2^h + 2^{\delta/2} - 2} \leq \frac{\delta}{2h} \qquad \text{, for even } \delta \leq d = h$$

held, for T heights h in the range from 2 to 100 (T of height 1 amounts to trivialities), and found the following results. There were only three failures, involving trees of heights 2 and 3.

Case II: odd $\delta > d$. (See Figure 4 for guidance.) Note that since h_d is a deepest node in H, $d \geq \lceil \delta/2 \rceil$. This time we consider the path of length d, consisting of ancestor
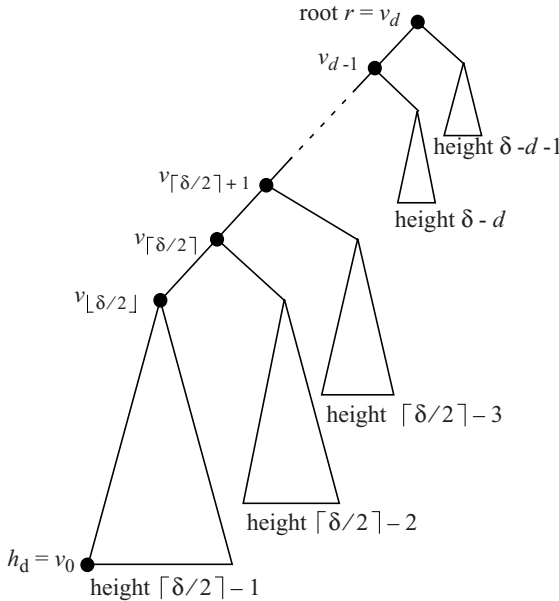

Fig. 3. δ is odd, $\delta > d$

nodes from h_d back to the root r of T; denote the nodes on this path as $h_d = v_0, v_1, v_2,$ ..., $v_d = r$. (See Figure 4.) Subtree T_H could contain all the nodes in a full subtree, rooted at $v_{\lfloor \delta/2 \rfloor}$ and having height $\lfloor \delta/2 \rfloor = \lceil \delta/2 \rceil - 1$. Similarly, T_H could contain all the nodes in a full subtree, rooted at the second child of $v_{\lfloor \delta/2 \rfloor + 1} = v_{\lceil \delta/2 \rceil}$, and having height $\delta - (\lfloor \delta/2 \rfloor + 2) = \lceil \delta/2 \rceil - 2$. Similarly for a full subtree, rooted at the second child of $v_{\lceil \delta/2 \rceil + 1}$, and having height $\lceil \delta/2 \rceil - 3$. And so on, back to a full subtree, rooted at the second child of v_{d-1}, and having height $\delta - d$; and, finally, a full subtree, rooted at the second child of T's root $r = v_d$, and having height $\delta - d - 1$. Thus, T_H could contain the $d - \lfloor \delta/2 \rfloor$ nodes $v_{\lceil \delta/2 \rceil}, ..., v_d$, plus the nodes in $d - \lfloor \delta/2 \rfloor + 1$ full subtrees of heights $\lceil \delta/2 \rceil - 1, ..., \delta - d - 1$. The number of nodes in T_H could be as great as

$$d - \lfloor \delta/2 \rfloor + ((2^{\lceil \delta/2 \rceil} - 1) + ... + (2^{\delta - d} - 1)) = 2^{\lceil \delta/2 \rceil + 1} - 2^{\delta - d} - 1.$$ The number of edges in T_H would be one less, or $2^{\lceil \delta/2 \rceil + 1} - 2^{\delta - d} - 2$. Reasoning as in Case I, we see that $dp(H)$ will be no greater than $rel\Delta(H)$ provided

$$\frac{2^{\lceil \delta/2 \rceil + 1} - 2^{\delta - d} - 2}{2^h - 1} \leq \frac{\delta}{2h} \quad , \text{ for odd } \delta > d \text{ when } d < h, \text{ and}$$

$$\frac{2^{\lceil \delta/2 \rceil + 1} - 2^{\delta - d} - 2}{2^h + 2^{\lfloor \delta/2 \rfloor} - 2} \leq \frac{\delta}{2h} \quad , \text{ for odd } \delta > d \text{ when } d = h.$$

At this point we can observe that in the event that the depth d of h_d equals the height h of T, and the diameter δ of H equals $2h - 1$, then the second inequality fails, since fraction $\dfrac{2^{\lceil \delta/2 \rceil + 1} - 2^{\delta - d} - 2}{2^h + 2^{\lfloor \delta/2 \rfloor} - 2}$ simplifies to 1, whereas $\dfrac{\delta}{2h} = \dfrac{2h - 1}{2h}$ is strictly less than 1.

A computer program which examined tree heights in the range from 2 to 100 revealed all the failure instances just commented upon. Beyond those, the run revealed only four other particular failures of the relation $dp(H) \leq rel\Delta(H)$ and they involved trees of heights 3, 4, and 5.

Let us contemplate the failures that are arising when $d = h$ and $\delta = 2h - 1$. We calculated the largest that T_H can be. T_H can achieve our bound, but only if H contains as fixed nodes all the nodes on the bottom levels of the full subtrees we cited, in which case H contains $2^{\lceil \delta/2 \rceil - 1} + 2^{\lceil \delta/2 \rceil - 2} + ... + 2^{\delta - d - 1} = 2^{\lceil \delta/2 \rceil} - 2^{\delta - d - 1}$ fixed nodes. When $d = h$ and $\delta = 2h - 1$, this means H contains $3 \cdot 2^{h-2}$ fixed nodes, which is slightly more than half of the $2^h + 2^{\lfloor \delta/2 \rfloor} - 2 = 3 \cdot 2^{h-1} - 2$ nodes in chromosome T. We generally expect building blocks to be smaller portions of the

chromosome than that. Put another way, the failures arising when $d = h$ and $\delta = 2h - 1$ are ones for quite atypically large schemas.

Case III: odd $\delta \le d$: Then $dp(H)$ will be no greater than $rel\Delta(H)$ provided

$$\frac{2^{\lceil \delta/2 \rceil + 1} - 3}{2^h - 1} \le \frac{\delta}{2h} \qquad \text{, for odd } \delta \le d < h \text{, and}$$

$$\frac{2^{\lceil \delta/2 \rceil + 1} - 3}{2^h + 2^{\lfloor \delta/2 \rfloor} - 2} \le \frac{\delta}{2h} \qquad \text{, for odd } \delta \le d = h$$

A computer program which examined if and when these inequalities held, for T heights h in the range from 2 to 100, revealed only three failures, involving trees of heights 2 and 3.

Case IV: even $\delta > d$: Then $dp(H)$ will be no greater than $rel\Delta(H)$ provided

$$\frac{3 \cdot 2^{\delta/2} - 2^{\delta - d} - 2}{2^h - 1} \le \frac{\delta}{2h} \qquad \text{, for even } \delta > d \text{ when } d < h \text{, and, and}$$

$$\frac{3 \cdot 2^{\delta/2} - 2^{\delta - d} - 2}{2^h + 2^{\delta/2} - 2} \le \frac{\delta}{2h} \qquad \text{, for even } \delta > d \text{ when } d = h.$$

We ran a computer program which examined if and when these inequalities held, for T heights h in the range from 2 to 100, with the following results. Invariably the first inequality failed when $h = d + 1$ and $\delta = 2d$. This class of failure, like that of Case II, arises when H contains a very large number of fixed nodes, approximately half the nodes of host chromosomal tree T. Beyond this class of failure, there were altogether only four other particular failures, and they involved trees of heights 3, 4, and 5.

Clearly there is a lesson to be learned from Propositions 2 and 3. The example tree given in the proof of Proposition 2 is irregular, a clump connected to a long string of bits, and for it $dp(H)$ can be made much greater than $rel\Delta(H)$. Proposition 3 shows us that $dp(H)$ is much better behaved when the binary tree is compact. If one is going to structure one's chromosome as a binary tree, then it is better to use as compact a tree as possible.

5   Conclusion

It is reasonable to say that schema disruption probability is better behaved the more closely it is related to how closely together are situated the fixed positions of the schema. That is, $dp(H)$ is better behaved the more closely it is approximated or dominated by $rel\Delta(H)$.

This paper contains two informative results, in Propositions 2 and 3. They show that if one is going to link the bits of a chromosome in the structure of a binary tree, then it is best to make the tree as compact as possible (in the context of the problem at hand).

The results of this paper are ones from a larger paper we are writing, in which we explore more types of trees and other issues as well. We will submit the larger paper elsewhere.

More generally we are interested in schema disruption behavior in chromosomes structured in other than the classical way as a linear sequence. After trees we plan to explore chromosomes that are 2- or 3- (or n-) dimensional grids. An example of the latter is to use as nodes the set $\{(x, y, z) \in \Re^3 \mid x, y, z$ are integers between 0 and $K\}$, then connect each node to each of its up to six neighbors in the axial directions. To cut such a chromosome in two, we could use random planes in 3-space, or random planes which are parallel to an axis, or some other way. These considerations await further exploration.

References

1. Holland, John (1975). *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI.

2. Goldberg, David E. (1989). *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Publishing, Reading, MA.

3. Greene, William A. (2000). "A Non-Linear Schema Theorem for Genetic Algorithms," in Whitley, D. (Eds.) Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2000), pp. 189-194. Morgan Kaufmann Publishers, San Francisco, CA.

4. Koza, John R. (1992). *Genetic Programming: On the Programming of Computers by Natural Selection*. MIT Press, Cambridge, MA

5. O'Reilly, Una-May (1995). *An Analysis of Genetic Programming*. PhD thesis, Carleton University, Ottawa-Carleton Institute for Computer Science, Ottawa, Ontario, Canada, 22 September 1995.

6. O'Reilly, Una-May, and Franz Oppacher (1995). "The Troubling Aspects of a Building Block Hypothesis for Genetic Programming", in Whitley, D. and Vose, M. D. (eds.) *Foundations of Genetic Algorithms 3*. Morgan Kaufmann Publishers, San Francisco.

7. Whigham, Peter A. (1995) "A Schema Theorem for Context-Free Grammars", in *1995 IEEE Conference on Evolutionary Computation*, Vol 1, pp. 178-181. IEEE Press

8. Rosca, Justinian P. (1997). "Analysis of Complexity Drift in Genetic Programming", in Koza, John R. *et al*. (eds) *Genetic Programming 1997: Proceedings of the Second Annual Conference* (pp. 286-294). Morgan Kaufmann Publishers, San Francisco, CA.

9. Poli, Riccardo, and William Langdon (1998). "Schema Theory for Genetic Programming with One-Point Crossover and Point Mutation". *Evolutionary Computation* 6(3), pp. 231-252. MIT Press, Cambridge, MA.