An Experimental Design Perspective on Genetic Algorithms

Colin Reeves and Christine Wright
Statistics and Operational Research Division
School of Mathematical and Information Sciences
Coventry University
UK
Email: CRReeves@cov.ac.uk

Abstract

In this paper we examine the relationship between genetic algorithms (GAs) and traditional methods of *experimental design*. This was motivated by an investigation into the problem caused by epistasis in the implementation and application of GAs to optimization problems: one which has long been acknowledged to have an important influence on GA performance. Davidor [1, 2] has attempted an investigation of the important question of determining the degree of epistasis of a *given* problem. In this paper, we shall first summarise his methodology, and then provide a critique from the perspective of experimental design. We proceed to show how this viewpoint enables us to gain further insights into the determination of epistatic effects, and into the value of different forms of encoding a problem for a GA solution. We also demonstrate the equivalence of this approach to the *Walsh transform* analysis popularized by Goldberg [3, 4], and its extension to the idea of *partition coefficients* [5]. We then show how the experimental design perspective helps to throw further light on the nature of deception.

1 INTRODUCTION

The term *epistasis* is used in the field of genetic algorithms to denote the effect on chromosome fitness of a combination of alleles which is not merely a linear function of the effects of the individual alleles. It can be thought of as expressing a degree of non-linearity in the fitness function, and roughly speaking, the more epistatic the problem is, the harder it may be for a GA to find its optimum.

Table 1: Goldberg's 3-bit deceptive function

String	Fitness
0 0 0	7
0 0 1	5
0 1 0	5
0 1 1	0
1 0 0	3
1 0 1	0
1 1 0	0
1 1 1	8

Several authors [3, 4, 6, 8] have explored the problem of epistasis in terms of the properties of a particular class of epistatic problems, those known as *deceptive* problems—the most famous example of which is probably Goldberg's 3-bit function, which has the form shown in Table 1 (definitions of this function in the literature may differ in unimportant details).

The study of such functions has been fruitful, but in terms of solving a given practical problem *ab initio*, it may not provide too much help. What might be more important would be the ability to estimate the degree of epistasis in a given problem before deciding on the most suitable strategy for solving it. At one end of the spectrum, a problem with very little epistasis should perhaps not be solved by a GA at all; for such problems one should be able to find a suitable linear or quasi-linear numerical method with which a GA could not compete. At the other end, a highly epistatic problem is unlikely to be solvable by any systematic method, including a GA. Problems with intermediate epistasis would be worth attempting with a GA, although even here it would also be useful if one could identify particular varieties of epistasis. If one could detect problems of a deceptive nature, for instance, one might suggest using an approach such as the 'messy GA' of [9, 10].

There is another aspect to this too: it is well-known (see e.g. [7, 11]) that the *coding* used for a GA may be of critical importance in how easy it is to solve. In fact (as we shall also demonstrate later) a particular choice of coding may render a simple linear function epistatic. Conversely, by choosing a different coding, it may be possible to reduce the degree of epistasis in a problem. It would clearly be valuable to be able to compare the epistasis existing in different codings of the same problem.

In recent papers, Davidor [1, 2] has reported an initial attempt at estimating the degree of epistasis in some simple problems. His results are to some degree perplexing, and it is difficult to draw firm conclusions from them. In this paper, we hope to show that his methodology can be put on a firmer footing by drawing on existing work in the field of *experimental design* (ED), which can be used to give insights into epistatic effects, and into the value of different codings. Later we shall also show how this approach relates to the Walsh transform methodology and the analysis of deception.

We begin by summarising Davidor's approach to the analysis of epistasis.

2 DAVIDOR'S EPISTASIS METHODOLOGY

Davidor deals with populations of binary strings $\{S\}$ of length l, for which he defines several quantities, as summarised below:

The basic idea of his analysis is that for a given population *Pop* of size N, the average fitness value can be determined as

$$\bar{V} = \sum_{S \in Pop} v(S)/N$$

where $v(S)$ is the fitness of string S. Subtracting this value from the fitness of a given string S produces the excess string fitness value

$$E(S) = v(S) - \bar{V}.$$

We may count the number of occurrences of allele a for each gene i, denoted by $N_i(a)$, and compute the average allele value

$$A_i(a) = \sum v(S)/N_i(a),$$

where the sum is over the strings whose i^{th} gene takes the value a. The excess allele value measures the effect of having allele a at gene i, and is given by

$$E_i(a) = A_i(a) - \bar{V}.$$

The genic value of string S is the value obtained by summing the excess allele values at each gene, and adding $\bar{V}$ to the result:

$$A(S) = \bar{V} + \sum_{i=1}^{l} E_i(a).$$

(Davidor actually gives the sum in the above formula the name 'excess genic value', i.e.

$$E(A) = \sum_{i=1}^{l} E_i(a),$$

although this quantity is not necessary in the ED context; we include the definition here for completeness.) Finally, the epistasis value is the difference between the actual value of string S and the genic value predicted by the above analysis:

$$\epsilon(S) = v(S) - A(S).$$

Thus far, what Davidor has done appears reasonably straightforward. He then defines further 'variance' measures, which he proposes to use as a way of quantifying the epistasis of a given problem. Several examples are given using some 3-bit problems, which demonstrate that using all 8 possible strings, his epistasis variance measure behaves in the expected fashion: it is zero for a linear problem, and increases in line with (qualitatively) more epistatic problems. However, when only a subset of the 8 possible strings is used, the epistasis measure gives rather problematic results, as evidenced by variances which are very hard to interpret.

In a real problem, of course, a sample of the 2^l possible strings is all we have, and an epistasis measure needs to be capable of operating in such circumstances. Below we re-formulate Davidor's analysis from an ED perspective, which we hope will shed rather more light on this problem.

3 AN EXPERIMENTAL DESIGN APPROACH

Davidor's analysis is complicated by the GA convention of describing a subset of strings as a *population*, when from a traditional statistical perspective it is actually a *sample*. Davidor uses the terms *Grand Population* and *sample population* to try to avoid this confusion. We propose instead to use the term *Universe* for the set of all possible 2^l strings, so that we can use the term population in the sense with which the GA community is familiar.

It is clear that Davidor is implicitly assuming an underlying linear model (defined on the bits) for the fitness of each string. This leads to a further problem in his analysis, linked to the above confusion between population and sample, in that he fails to distinguish between the *parameters* of this underlying model, and the *estimates* of those parameters which are possible for a given population. We can begin to explain this more clearly by first making the model explicit.

We can express the full epistatic model as

$$\begin{aligned} v(S) \quad = \quad & \text{constant} + \sum_{i=1}^{l} (\text{effect of allele at gene } i) \\ & + \sum_{i=1}^{l-1} \sum_{j=i+1}^{l} (\text{interaction between alleles at gene } i \text{ and gene } j) \\ & + \ldots \\ & + (\text{interaction between alleles at gene 1, gene 2, } \ldots, \text{ gene } l) \\ & + \text{random error} \end{aligned}$$

In conventional experimental design, the above model would actually be written in parametric form. For example, the model for a string of 3 binary bits could be written as follows:

$$v_{pqrs} = \mu + \alpha_p + \beta_q + (\alpha\beta)_{pq} + \gamma_r + (\alpha\gamma)_{pr} + (\beta\gamma)_{qr} + (\alpha\beta\gamma)_{pqr} + \varepsilon_{pqrs} \qquad (1)$$

where v_{pqrs} is the fitness of the string (p, q, r), and the subscript s denotes the replication number (i.e. the s^{th} occurrence of the string). If there is no intrinsic noise, we can of course drop the subscript s. The parameters on the right-hand side are as follows:

μ	average fitness
α_p	effect of allele p at gene 1
β_q	effect of allele q at gene 2
$(\alpha\beta)_{pq}$	joint effect of allele p at gene 1 and allele q at gene 2
γ_r	effect of allele r at gene 3
$(\alpha\gamma)_{pr}$	joint effect of allele p at gene 1 and allele r at gene 3
$(\beta\gamma)_{qr}$	joint effect of allele q at gene 2 and allele r at gene 3
$(\alpha\beta\gamma)_{pqr}$	joint effect of allele p at gene 1, allele q at gene 2 and allele r at gene 3
ε_{pqrs}	random error for replication s of string (p, q, r)

Davidor assumes zero random error, which is reasonable in many, although not all, applications of GAs. We thus intend to ignore the possibility of random error here, although we hope to consider such problems at a later date.

We emphasize again that we must distingush two different situations, even when we assume zero random error. In the first case we know the fitness of every string in the Universe. In

practice this is unrealistic—in reality we only know the fitness of every string in a subset of the Universe (i.e. our 'population', to use the conventional GA terminology, is merely a sample). Of course, in the first case, there is in one sense no problem: the optimal combination is obvious, and all the measures proposed by Davidor are constants. In the second case (which is the real situation) the various epistasis measures are only *estimates* of parameters, whose expectations and variances are important characteristics. Nevertheless, for purposes of exposition, we need to focus initially on the first case, and we shall postpone examination of the real situation to another paper.

3.1 An example

Suppose we have a 3-bit string, and the fitness of every string in the Universe is known. There are of course $2^3 = 8$ strings , and therefore 8 fitness values, but the experimental design model above has 27 parameters. It is thus essential to impose some side conditions if these parameters are to be estimated; the usual ones are the obvious constraints that at every order of interaction, the parameters sum to zero for each subscript. This results in an additional 19 independent relationships such as

$$\begin{aligned} \sum_p \alpha_p &= 0 \\ \sum_p (\alpha\beta)_{pq} &= 0 \quad \text{for } q = 0, 1 \\ \sum_p (\alpha\beta\gamma)_{pqr} &= 0 \quad \text{for } q, r = 0, 1 \end{aligned}$$

and thus allows the 'solution' of the above model, in the sense that all the parameter values can be determined if we have observed every one of the 8 possible strings—the first case above. For example, we find that

$$\begin{aligned} \mu &= v_{***} \\ \mu + \alpha_p &= v_{p**} \quad \text{for } p = 0, 1 \\ \mu + \beta_q &= v_{*q*} \quad \text{for } q = 0, 1 \\ \mu + \gamma_r &= v_{**r} \quad \text{for } r = 0, 1 \end{aligned}$$

where the notation v_{p**}, for instance, means averaging over subscripts q and r. The effects can be seen to be exactly equivalent to Davidor's 'excess allele values' as defined above. For instance, his $A_1(p) = v_{p**}$, so that $E_1(p) = \alpha_p$. Similarly, his 'excess genic values' $E(A)$ are found by summing α_p, β_q and γ_r for each possible combination of p, q, r. Finally, his 'string genic value' is clearly

$$\mu + \alpha_p + \beta_q + \gamma_r .$$

The difference between the actual value and the genic value, $\epsilon(S)$, is therefore simply the sum of all the interaction terms. If there is no epistasis, then by definition the combinations of alleles p, q, r will have no effect on chromosome fitness other than this simple linear sum, so that epistasis can be interpreted as the combined effect of the interaction terms.

3.2 Analysis of Variance

The normal procedure in experimental design is to perform an 'Analysis of Variance' (Anova), whereby the variability of the fitness values (measured by sums of squared deviations from mean fitness, and denoted by SS) is partitioned into orthogonal components

from identifiable sources. In the table below, we give a conventional Anova table for our 3-bit example, with Davidor's notation alongside:

Table 2: Analysis of Variance Table

Source of variation	Degrees of freedom	Sum of squares (SS)	Davidor's notation
Between alleles of gene 1	1	$\sum_{pqr}(v_{p**} - v_{***})^2$	$\sum[E_1(a)]^2$
Between alleles of gene 2	1	$\sum_{pqr}(v_{*q*} - v_{***})^2$	$\sum[E_2(a)]^2$
Between alleles of gene 3	1	$\sum_{pqr}(v_{**r} - v_{***})^2$	$\sum[E_3(a)]^2$
Total main effects (i.e.'genic' effect)	3	sum of above	sum of above
Interactions (i.e.epistatic effect)	4	$\sum_{pqr}(v_{pqr} - v_{p**} - v_{*q*} - v_{**r} + 2v_{***})^2$	$\sum[\epsilon(S)]^2$
Total	7	$\sum(v_{pqr} - v_{***})^2$	$\sum[E(S)]^2$

The *degrees of freedom* are the number of independent elements in the associated SS; for example, in the Total SS term, only 7 of the $(v_{pqr} - v_{***})$ terms are independent, since they must satisfy the relationship $\sum_{pqr}(v_{pqr} - v_{***}) = 0$.

It is well-known (and easy to prove) that

$$\text{Total SS} = \text{Main effects SS} + \text{Interactions SS}$$

and since Davidor has simply divided these values by a constant to obtain his 'variances', it is hardly surprising that he finds that

$$\text{Total 'variance'} = \text{Genic 'variance'} + \text{Epistasis 'variance'}.$$

(We note here that when we come to investigate the real situation, we shall see that this result *appears* no longer to be true using Davidor's definitions; the reason for this will be discussed in the second paper.)

Any standard statistical computing package will produce these Anova tables; below we give some examples obtained using MINITAB on Davidor's functions f_1, f_2, f_3 and f_4. These functions represent respectively a linear function, a delta function, a mixture of f_1 and f_2, and finally the deceptive function of Table 1.

We see from these results that in a qualitative sense (for these functions at least), the amount of epistasis can be inferred from the relative magnitudes of the SS terms, i.e. the SS values as a fraction of the total SS. In case f_1, the Anova table shows no epistasis at all, as would be expected, while f_3 appears to be much less epistatic than f_2. The case of f_4 (the deceptive function) is interesting: the *relative* magnitude of interactions SS is much greater than in the case of f_2 (the delta function)—that is, it is worse to have *misleading* information than to have no information at all. (We note here that Davidor [1] interprets the cases of f_2 and f_4 differently—arguing from the actual *numerical values* of his 'epistasis

Table 3: Anova results for Davidor's functions

	f_1		f_2		f_3		f_4	
Source	df	SS	df	SS	df	SS	df	SS
Main effects	3	42.00	3	294.00	3	133.00	3	5.50
2-way interactions	3	0.00	3	294.00	3	73.50	3	44.00
3-way interactions	1	0.00	1	98.00	1	24.50	1	24.50
Total	7	42.00	7	686.00	7	231.00	7	74.00

variances' that the deceptive function is *less* epistatic than f_2. However, this would imply that epistasis is dependent on the measurement scale of the function, whereas it is clear that this should not influence the performance of a GA. We believe therefore that looking at the relative magnitudes in the Anova table is more informative. Whether we can then go on to infer that this indicator of epistasis necessarily means that the problem is hard for a GA to solve is of course a separate, although very important, issue—one to which we hope to return in a future paper.)

4 THE INFLUENCE OF CODING

Experimental design also helps to throw some light on the often-noticed influence of the adopted coding on the ease or difficulty of solving a given problem using GAs. We now consider 2 cases that have attracted attention in the GA literature: the influence of Gray coding, and the effect of using a binary rather than a q-ary alphabet ($q > 2$).

4.1 Gray coding

Another of Davidor's functions is a Gray-coded version of his function f_1. The case for Gray coding has been put persuasively by Caruana and Schaffer [16], but Davidor's example in [2] shows that it may not necessarily be helpful.

Consider the two representations of a 3-bit problem as tabulated below:

Table 4: Binary and Gray code versions of a 3-bit problem

Binary representation	Fitness value	Gray representation
0 0 0	v_1	0 0 0
0 0 1	v_2	0 0 1
0 1 0	v_3	0 1 1
0 1 1	v_4	0 1 0
1 0 0	v_5	1 1 0
1 0 1	v_6	1 1 1
1 1 0	v_7	1 0 1
1 1 1	v_8	1 0 0

A useful experimental design concept here is that of a *contrast*, usually denoted by upper case Roman letters. For example, the contrast

$$A = \alpha_1 - \alpha_0$$

(where α_p is as previously defined), expresses the average fitness value when allele 1 is instantiated at gene 1, compared to the instantiation of allele 0. In terms of the vector of fitness values $\mathbf{v}$ in binary representation from the table above,

$$A = \frac{1}{4}(-1,-1,-1,-1,1,1,1,1)\mathbf{v}$$

$$B = \frac{1}{4}(-1,-1,1,1,-1,-1,1,1)\mathbf{v}$$

$$C = \frac{1}{4}(-1,1,-1,1,-1,1,-1,1)\mathbf{v}.$$

Similarly, we can define contrasts relating to the interaction effects, so that

$$AB = \frac{1}{4}(1,1,-1,-1,-1,-1,1,1)\mathbf{v}$$

expresses the average fitness value for cases where the instantiated alleles at genes 1 and 2 are the same, compared to those where they are different. The other contrasts are as follows:

$$AC = \frac{1}{4}(1,-1,1,-1,-1,1,-1,1)\mathbf{v}$$

$$BC = \frac{1}{4}(1,-1,-1,1,1,-1,-1,1)\mathbf{v}$$

$$ABC = \frac{1}{4}(-1,1,1,-1,1,-1,-1,1)\mathbf{v}.$$

The contrast ABC can be regarded as the difference between AB with allele 1 instantiated at gene 3 and AB with allele 0 at gene 3. (Alternatively ABC could be interpreted in terms of AC or BC.) These 7 contrasts are each associated with 1 degree of freedom, and correspond to the information presented in Table 2; they are orthogonal, and can thus be determined simultaneously from the observed fitness values. In the case of Davidor's f_1, for example, they are $A = 4, B = 2, C = 1$ and all others 0.

Now consider the Gray-coded version of the same situation, where we denote the contrasts by the letters X,Y,Z. While it is clear that

$$X \equiv A,$$

the other contrasts are all different: for example,

$$Y = \frac{1}{4}(-1,-1,1,1,1,1,-1,-1)\mathbf{v},$$

so that

$$Y \equiv -AB.$$

Similar results can be found for the other contrasts, which can be summarised as follows:

$$A \equiv X, B \equiv -XY, C \equiv XYZ, AB \equiv -Y, AC \equiv YZ, BC \equiv -Z, ABC \equiv -XZ.$$

Thus, analysing Davidor's linear function f_1 using the above Gray code representation would result in non-zero contrasts for the interactions XY and XYZ, and a conclusion from the Anova table that the function was epistatic. Of course, it would not be difficult to define a function for which a Gray code had the opposite effect—the 3-bit function displayed in Table 5 below is epistatic, but it is not difficult to show that using the Gray code of Table 4 would make the problem linear.

Table 5: Another 3-bit problem

String (binary code)	Fitness
0 0 0	8
0 0 1	5
0 1 0	1
0 1 1	4
1 0 0	5
1 0 1	2
1 1 0	6
1 1 1	9

We also note here a connection with the work of Liepins and Vose [6], who show that there is always a transformation of the coding of a 'fully deceptive' problem which transforms it into a 'fully easy' one (for a definition of these terms see [6]). In this sense, a Gray code transformation of a binary code is simply a special case of their more general result. In terms of experimental design, what they are saying is that there is always a way of converting interactions into main effects by a suitable transformation. The problem in practice, of course, is to know what that transformation is!

4.2 Binary versus q-ary coding

The issue of whether binary coding is to be preferred to using a larger q-ary alphabet ($q > 2$) has been widely debated, and it would be fair to say that it has not been resolved. Holland [14], and following him Goldberg [15], stressed the advantage of a binary alphabet, in that it allows the sampling of the maximum number of schemata per individual in the population. More recently, Antonisse has put forward a counter-argument in [17] by re-defining the concept of a schema, while Radcliffe's work [11] makes a very similar point. On the other hand, Reeves [18] has recently argued that there are certain theoretical advantages in using binary-coding in cases where GAs need to be limited to a small number of function evaluations. An ED approach throws a further interesting sidelight on the question.

Suppose we have a problem with 2 genes J and K, each of which has 4 alleles denoted by {0,1,2,3}. Then, defining the fitness vector as

$$\mathbf{v} = (v_{00}, v_{10}, \ldots, v_{33})^T,$$

we can determine 3 orthogonal contrasts for each gene

$$J_1 = (-1,-1,1,1,-1,-1,1,1,-1,-1,1,1,-1,-1,1,1)\mathbf{v},$$

$$J_2 = (1,-1,-1,1,1,-1,-1,1,1,-1,-1,1,1,-1,-1,1)\mathbf{v},$$

$$J_3 = (-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1)\mathbf{v},$$

and

$$K_1 = (-1,-1,-1,-1,-1,-1,-1,-1,1,1,1,1,1,1,1,1)\mathbf{v},$$

$$K_2 = (1,1,1,1,-1,-1,-1,-1,-1,-1,-1,-1,1,1,1,1)\mathbf{v}.$$

$$K_3 = (-1,-1,-1,-1,1,1,1,1,-1,-1,-1,-1,1,1,1,1)\mathbf{v},$$

The interpretation of these contrasts is a little more complicated than in the binary case, but it can easily be seen that J_1, for example, expresses the contrast between having alleles at 'high' levels at gene 1 rather then at 'low' levels. We could thus interpret J_1 (and, naturally, K_1) as indicating a 'linear' component, while the pattern of positive and negative signs for J_2, K_2 and J_3, K_3 suggest 'quadratic' and 'cubic' components respectively.

Suppose for a particular $\mathbf{v}$ the main effects give the only non-zero contrasts using this coding. For example, suppose the fitness is defined as

$$v_{jk} = 1 + 2j + k \qquad \text{for } j, k \in \{0, 1, 2, 3\}.$$

Consider what happens if the 4-ary code $\{jk\}$ is replaced by its binary equivalent $\{pqrs\} = (0000, 0100, 1000, \ldots, 1111)$. There will now be 4 genes P,Q,R,S leading to the following contrasts

$$P = (-1,-1,1,1,-1,-1,1,1,-1,-1,1,1,-1,-1,1,1)\mathbf{v},$$
$$Q = (-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1,-1,1)\mathbf{v},$$
$$R = (-1,-1,-1,-1,-1,-1,-1,-1,1,1,1,1,1,1,1,1)\mathbf{v},$$
$$S = (-1,-1,-1,-1,1,1,1,1,-1,-1,-1,-1,1,1,1,1)\mathbf{v}.$$

These are seen to be identical to J_1, J_3, K_1, K_3 respectively, but what has happened to J_2 and K_2? It is in fact easily checked that the information contained in J_2 and K_2 will now be found in the contrasts PQ and RS, so that an Anova table based on a binary coding of such a function would again lead to the conclusion that the problem is epistatic.

In contrast to the binary versus Gray question, it would seem more doubtful that adoption of a binary coding could make an epistatic q-ary problem less so. Thus, to the extent that it is harder for a GA to solve an epistatic problem than a simple linear one (and we note that in the latter case we would not actually need to use a GA at all), we might argue that binary coding of the function is likely to increase epistasis, so that any supposed advantage from binary coding could be negated.

5 WALSH TRANSFORMS AND DECEPTION

Thus far we have seen that Davidor's linear decomposition of a bit-encoded function leads to a set of coefficients which are equivalent to the standard linear model of experimental design. Another linear decomposition which is often used in the analysis of GAs is the *Walsh transform*.

Bethke [19] introduced the idea of using Walsh transforms to analyse the process of a GA in the case of binary-coded strings. The ideas used were given greater impetus and wider currency in papers by Goldberg [3, 4]. More recently, Mason [5] has defined the concept of a *partition coefficient* as a generalization of the Walsh coefficients for non-binary strings. He proceeds to derive some theoretical results from this definition, which makes it clear that these coefficients are just the 'effects' as defined in the ED context, and his theoretical results are simply a derivation of the side constraints as outlined above. It further follows from this that the Walsh transform decomposition is also equivalent to that of experimental design. However, it is instructive to examine the relationship between Walsh transform analysis and experimental design rather more closely. We shall focus particularly on Goldberg's famous 3-bit deceptive problem, as in Table 1.

In Walsh transform analysis, the bits are usually numbered from right to left, so *in this section only* we shall adopt the same convention. The Walsh monomials are defined on the string positions $\{y_i\}$ coded for convenience as $+1$ or -1 rather than the usual 0 or 1:

$$\psi_j(y) = \Pi_{i=1}^{l}(y_i)^{j_i}$$

where j_i is the i^{th} bit (counting from the right) in the binary representation of the number j. The Walsh function representation of the fitness v is

$$v(y) = \sum_{j=0}^{2^l-1} w_j \psi_j(y)$$

where y encodes the bit positions as above. There are clearly the same number of independent coefficients in the ED decomposition as there are Walsh coefficients, so it is natural to ask how they are related.

The relationship is clearly illustrated in a 3-bit example. The Walsh coefficients can be found from the fitness averages for different schemata:

$$\begin{aligned} v_{***} &= w_0 \\ v_{**0} &= w_0 + w_1 \\ v_{**1} &= w_0 - w_1 \quad \text{etc} \end{aligned}$$

whereas from the experimental design viewpoint, we have

$$\begin{aligned} v_{***} &= \mu \\ v_{**0} &= \mu + \alpha_0 \\ v_{**1} &= \mu + \alpha_1 \quad \text{etc.} \end{aligned}$$

If we write out the full set of equations, we find that

$$\begin{aligned} \mu &= w_0 \\ \alpha_i &= (-1)^i w_1 \\ \beta_j &= (-1)^j w_2 \\ (\alpha\beta)_{ij} &= (-1)^{i+j} w_3 \\ \gamma_k &= (-1)^k w_4 \\ (\alpha\gamma)_{ik} &= (-1)^{i+k} w_5 \\ (\beta\gamma)_{jk} &= (-1)^{j+k} w_6 \\ (\alpha\beta\gamma)_{ijk} &= (-1)^{i+j+k} w_7 \end{aligned}$$

The 'mapping' from the Walsh coefficient numbers to the appropriate 'effect' is given by writing the effects in what is known in experimental design as *standard order*: in this case $\{\mu, \alpha, \beta, \alpha\beta, \gamma, \alpha\gamma, \beta\gamma, \alpha\beta\gamma\}$. The general pattern is fairly obvious—on adding another factor the next set of effects is obtained by 'combining' the new factor with the effects already listed, in the same order. Thus in the case of a 4-bit problem, for example, the next 8 effects in standard order will be

$$\{\delta, \alpha\delta, \beta\delta, \alpha\beta\delta, \gamma\delta, \alpha\gamma\delta, \beta\gamma\delta, \alpha\beta\gamma\delta\}.$$

It is also fairly obvious that this order is a consequence of the definition of the Walsh monomials.

Thus in general, to convert from the Walsh representation to the ED coefficients, we first identify the appropriate coefficient as above, and its associated indices, and then multiply by $(-1)^{\sum \text{indices}}$.

5.1 Implications for deception

In his first paper [3], Goldberg uses Walsh coefficients to design the fully deceptive 3-bit function of Table 1. The requirement for this function is that while 111 is the optimal point, any schema containing 1s should be less fit than the corresponding schema which contains 0s: for example, $v_{**1} < v_{**0}$. We now consider this function from the ED viewpoint.

For example, the inequality $v_{**1} < v_{**0}$ can be decomposed as follows (remembering that the numbering is from right to left, so that the specified gene here corresponds to α). $v_{**1} < v_{**0}$ implies that

$$v_{001} + v_{011} + v_{101} + v_{111} < v_{000} + v_{010} + v_{100} + v_{110}.$$

On substituting the ED model given in Equation 1, the left-hand-side of this inequality is

$$4\mu + 4\alpha_1 + 2[\beta_0 + \beta_1 + \gamma_0 + \gamma_1] + 2[(\alpha\beta)_{10} + (\alpha\beta)_{11}] + 2[(\alpha\gamma)_{10} + (\alpha\gamma)_{11}] + (\beta\gamma)_{00} + (\beta\gamma)_{01} + (\beta\gamma)_{10} + (\beta\gamma)_{11} + (\alpha\beta\gamma)_{100} + (\alpha\beta\gamma)_{110} + (\alpha\beta\gamma)_{101} + (\alpha\beta\gamma)_{111}$$

while the right-hand-side is

$$4\mu + 4\alpha_0 + 2(\beta_0 + \beta_1 + \gamma_0 + \gamma_1) + 2[(\alpha\beta)_{00} + (\alpha\beta)_{01}] + 2[(\alpha\gamma)_{00} + (\alpha\gamma)_{01}] + (\beta\gamma)_{00} + (\beta\gamma)_{01} + (\beta\gamma)_{10} + (\beta\gamma)_{11} + (\alpha\beta\gamma)_{000} + (\alpha\beta\gamma)_{010} + (\alpha\beta\gamma)_{001} + (\alpha\beta\gamma)_{011}.$$

Many of these terms cancel, while because of the side constraints terms such as $(\alpha\beta)_{10} + (\alpha\beta)_{11}$ vanish, and we are simply left with

$$\alpha_1 < \alpha_0.$$

The other order-1 schemata inequalities similarly reduce to

$$\beta_1 < \beta_0, \gamma_1 < \gamma_0$$

which, again because of the side constraints, simply mean that the effects with the '1' subscripts are the negative ones. Thus we could write

$$\alpha_1 = -a, \alpha_0 = a, \quad \text{etc}$$

where it is to be understood that $a > 0$. It can also be shown that the order-2 inequalities lead to relationships of the form

$$\begin{aligned} \alpha_1 + (\alpha\beta)_{10} &< \alpha_0 + (\alpha\beta)_{00} \\ \beta_1 + (\alpha\beta)_{01} &< \beta_0 + (\alpha\beta)_{00} \\ \alpha_1 + \beta_1 + (\alpha\beta)_{11} &< \alpha_0 + \beta_0 + (\alpha\beta)_{00} \end{aligned}$$

The first two constraints reduce to

$$\begin{aligned} a + (ab) &> 0 \\ b + (ab) &> 0 \quad \text{etc} \end{aligned}$$

where, because of the side constraints,

$$\begin{aligned} (\alpha\beta)_{00} &= (\alpha\beta)_{11} = (ab) \\ (\alpha\beta)_{01} &= (\alpha\beta)_{10} = -(ab), \quad \text{etc.} \end{aligned}$$

The third constraint is redundant, as the interaction terms cancel.

Finally, we have the fact that v_{111} is the optimum, leading to 7 inequalities generated by $v_{111} > v_{011}$ etc. After some algebra, these reduce to the following:

$$\begin{aligned}
(ab) + (ac) &> b + c \\
(ab) + (ac) &> a + (abc) \\
(ab) + (bc) &> a + c \\
(ab) + (bc) &> b + (abc) \\
(ac) + (bc) &> a + b \\
(ac) + (bc) &> c + (abc) \\
(abc) &< -(a + b + c)
\end{aligned}$$

The last inequality puts an upper bound on the third-order interaction, (abc), and also forces it to be negative. The other conditions occur in pairs, each of them having the following interpretations:

- for each factor, the sum of the interactions with the other two factors must exceed the sum of the other two main effects;
- for each factor, the sum of the interactions with the other two factors and the third-order interaction must exceed that main effect (where we have used the fact that (abc) is negative).

There are two comments here: firstly it is interesting that deception corresponds to 'large' interaction terms. There is a possible link here with the results of Liepins and Vose [6] who, although using yet another decomposition, found similar conditions for distinguishing between levels of epistasis. (It is obviously possible, although perhaps less interesting, to relate their polynomial decomposition to experimental design. The conditions on the coefficients in their decomposition do not have as 'nice' an interpretation as the above.)

The second comment relates to the relative transparency of this way of expressing the deception conditions. We would argue that they are rather more meaningful than when they are expressed by the rather anonymous Walsh coefficients. In fact, this analysis revealed an error in the specification given in Goldberg [3]—probably due to a typographical mistake which would be much harder to overlook using the ED formulation[1]. Remarkably, on comparing the ED decomposition to the Liepins and Vose representation, it was clear that there was also an error in one of the definitions in [6]!

6 CROSSOVER NON-LINEARITY RATIOS

Earlier, we referred to Mason's extension [5] of the Walsh transform decomposition to what he calls partition coefficients in the general (non-binary) case. These he denotes by symbols such as $\epsilon(i**)$, which in ED terms represents the effect of allele i at gene 1. That is, his $\epsilon(i**)$ is just the term we have called α_i.

In a more recent paper [21], Mason has taken this concept a stage further in an attempt to analyse the effect of traditional crossover and how this operator interacts with a given function. This is an important question, as it marks a step beyond the essentially static

[1] Goldberg [20] has confirmed that two inequalities which *should* read $w_3 + w_5 > w_1 + w_7$ and $w_3 + w_6 > w_2 + w_7$ have had their right-hand sides transposed in [3].

analysis of epistasis to a consideration of dynamic aspects. In Mason's terminology, if two strings ab and pq are crossed to produce aq and pb, where a, b, p, q may all represent *sub-strings* of several bits, we can form a *crossover non-linearity ratio*

$$\psi = \frac{|\epsilon(ab)|}{sign[\epsilon(a*)\epsilon(*b)][|\epsilon(a*)| + |\epsilon(*b)|]},$$

where the $\epsilon(a*)$ are now 'pseudo partition coefficients'. The purpose of this is to attempt to identify cases where crossover is likely to fail to combine building blocks usefully.

Unfortunately, he makes the assumptions that $\epsilon(a*) = -\epsilon(p*), \epsilon(*b) = -\epsilon(*q)$ etc. These relations are perfectly valid in the case where a, b, p, q represent single bits, but it does not follow when they represent several bits. We can see this quite easily from the ED viewpoint, if we take the simplest non-trivial case of 3-bit binary strings, where a, p represent the first 2 bits, and b, q the last one.

Using the ED decomposition of Equation 1, we can identify Mason's pseudo partition coefficients as follows (in an obvious notation):

$$\begin{aligned}
\epsilon(a*) &= \alpha_{i_a} + \beta_{j_a} + (\alpha\beta)_{i_a j_a} \\
\epsilon(*b) &= \gamma_{k_b} \\
\epsilon(ab) &= (\alpha\gamma)_{i_a k_b} + (\beta\gamma)_{j_a k_b} + (\alpha\beta\gamma)_{i_a j_a k_b}
\end{aligned}$$

Assuming that a, p are not identical, it is clear we have two cases to consider. If *both* $i_a \neq i_p$ and $j_a \neq j_p$ then

$$\epsilon(a*) + \epsilon(p*) = 2(\alpha\beta)_{i_a j_a} \neq 0$$

because of the side constraints. Similarly, if just one of $i_a = i_p$ or $j_a = j_p$ is true, then we have

$$\epsilon(a*) + \epsilon(p*) = 2\alpha_{i_a} \neq 0 \quad \text{or} \quad \epsilon(a*) + \epsilon(p*) = 2\beta_{j_a} \neq 0$$

so that in neither case does the single-bit result follow through. There must consequently be some doubt as to the usefulness of ψ: a value of near zero is interpreted in [21] as indicating low epistasis and thus a situation where traditional one-point crossover is likely to be effective. However, it is clear from the above decomposition of his $\epsilon(ab)$ that a zero value of the ψ ratio could result from an appropriate combination of interaction terms of different orders.

7 CONCLUSIONS

We have shown that there are considerable and interesting links between genetic algorithms and traditional experimental design methods, and that ED can help to illuminate the still inadequately understood nature of epistasis in GAs. These links have been adumbrated and explored in the context of three applications in the GA literature: Davidor's 'epistasis variance'; the Walsh transform analysis of Goldberg; and Mason's attempt to extend the latter to investigate the interaction between the characteristics of a function and the crossover operator. In each case, the ED perspective is helpful; it provides another way of formulating and understanding what the existing methodology is doing—a way which we would argue is more transparent and intuitive.

However, this approach has in common with existing methodology that it begs a very large question: in practice we have no knowledge of the Universe. This means that measures

of epistasis, for instance, which assume such knowledge may give unpredictable and even contradictory results when we base them on sample information. In fact, experimental design has a long history of dealing with this problem, and in a further paper, currently in preparation, we hope to show how light can be thrown on this crucial question by drawing on the 50 years of experience which statisticians have accumulated in using experimental design. As already mentioned, it is also as yet far from certain whether the epistasis measures that have been developed actually do indicate cases which are in practice hard or easy for a GA (or indeed any other heuristic), but we hope that the ED approach will also enable this question to be more carefully addressed.

In summary, we believe that the experimental design perspective on GAs has much to commend it. At the very least it gives GA researchers another tool for approaching the analysis of GA performance. The history of the past decade has been of exciting and novel developments of genetic algorithms which have somewhat outstripped the development of tools for thinking theoretically about what GAs are doing. We hope that in a small way this paper will give the GA community something else to help in this endeavour.

References

[1] Y.Davidor (1990) Epistasis variance: suitability of a representation to genetic algorithms. *Complex Systems*, **4**, 369-383.

[2] Y.Davidor (1991) Epistasis variance: a viewpoint on GA-hardness. *In* G.J.E.Rawlins (Ed.) (1991) *Foundations of Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.

[3] D.E.Goldberg (1989) Genetic algorithms and Walsh functions: part I, a gentle introduction. *Complex Systems*, **3**, 129-152.

[4] D.E.Goldberg (1989) Genetic algorithms and Walsh functions: part II, deception and its analysis. *Complex Systems*, **3**, 153-171.

[5] A.J.Mason (1991) Partition coefficients, static deception and deceptive problems for non-binary alphabets. *In* [23], 210-214.

[6] G.E.Liepins and M.D.Vose (1990) Representational issues in genetic optimization. *J.Exper. and Theor. Artificial Intelligence*, **2**, 101-115.

[7] M.D.Vose and G.E.Liepins (1991) Schema disruption. *In* [23], 237-242.

[8] D.Whitley(1992) Deception, dominance and implicit parallelism in genetic search. *Annals of Maths. and AI*, **5**, 49-78.

[9] D.E.Goldberg, B.Korb and K.Deb (1989) Messy genetic algorithms: motivation, analysis and first results. *Complex Systems*, **3**, 493-530.

[10] D.E.Goldberg, K.Deb and B.Korb (1990) Messy genetic algorithms revisited: studies in mixed size and scale. *Complex Systems*, **4**, 415-444.

[11] N.J.Radcliffe (1992) Non-linear genetic representations. *In* R.Männer and B.Manderick (Eds.) (1992) *Parallel problem-Solving from Nature 2.* Elsevier Science Publishers, Amsterdam.

[12] O.Kempthorne (1952) *The Design and Analysis of Experiments.* Wiley, New York.

[13] D.C.Montgomery (1991) *Design and Analysis of Experiments.* Wiley, New York.

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

[15] D.E.Goldberg (1989) *Genetic Algorithms in Search, Optimization, and Machine Learning.* Addison-Wesley, Reading, Mass.

[16] R.A.Caruana and J.D.Schaffer (1988) Representation and hidden bias: Gray vs. binary coding for genetic algorithms. In *Proc. 5th International Conference on Machine Learning.* Morgan Kaufmann, Los Altos, CA.

[17] J.Antonisse (1989) A new interpretation of schema notation that overturns the binary encoding constraint. *In* [22], 86-91.

[18] C.R.Reeves (1993) Using genetic algorithms with small populations. *In* [24], 92-99.

[19] A.D.Bethke (1981) *Genetic Algorithms as Function Optimizers.* Doctoral dissertation, University of Michigan.

[20] D.E.Goldberg (1993) Personal communication.

[21] A.J.Mason (1993) *Crossover Non-linearity Ratios and the Genetic Algorithm: Escaping the Blinkers of Schema Processing and Intrinsic Parallelism.* Report No. 535b, School of Engineering, University of Auckland, NZ.

[22] J.D.Schaffer (Ed.) (1989) *Proceedings of* 3^{rd} *International Conference on Genetic Algorithms.* Morgan Kaufmann, Los Altos, CA.

[23] R.K.Belew and L.B.Booker (Eds.) (1991) *Proceedings of* 4^{th} *International Conference on Genetic Algorithms.* Morgan Kaufmann, San Mateo, CA.

[24] S.Forrest (Ed.) (1993) *Proceedings of* 5^{th} *International Conference on Genetic Algorithms*, Morgan Kaufmann, San Mateo, CA.