

Fluctuating Crosstalk, Deterministic Noise, and GA Scalability

Paul Winward

Illinois Genetic Algorithms Laboratory
Department of General Engineering
University of Illinois at Urbana-Champaign
117 Transportation Building
104 S. Mathews Avenue, Urbana, IL 61801
winward@uiuc.edu

David E. Goldberg

Illinois Genetic Algorithms Laboratory
Department of General Engineering
University of Illinois at Urbana-Champaign
117 Transportation Building
104 S. Mathews Avenue, Urbana, IL 61801
deg@uiuc.edu

ABSTRACT

This paper extends previous work showing how fluctuating crosstalk in a deterministic fitness function introduces noise into genetic algorithms. In that work, we modeled fluctuating crosstalk or nonlinear interactions among building blocks via higher-order Walsh coefficients. The fluctuating crosstalk behaved like exogenous noise and could be handled by increasing the population size and run duration. This behavior held until the strength of the crosstalk far exceeded the underlying fitness variance by a certain factor empirically observed. This paper extends that work by considering fluctuating crosstalk effects on genetic algorithm scalability using smaller-ordered Walsh coefficients on two extremes of building block scaling: uniformly-scaled and exponentially-scaled building blocks. Uniformly-scaled building blocks prove to be more sensitive to fluctuating crosstalk than do exponentially-scaled building blocks in terms of function evaluations and run duration but less sensitive to population sizing for large building-block interactions. Our results also have implications for the relative performance of building-block-wise mutation over crossover.

Categories and Subject Descriptors

I.2.8 [Artificial Intelligence]: Problem Solving, Control Methods, and Search; G.1.6 [Numerical Analysis]: Optimization

General Terms

Algorithms, Performance

Keywords

Genetic algorithm, fluctuating crosstalk, deterministic noise, scalability, problem difficulty

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

GECCO '06, July 8–12, 2006, Seattle, Washington, USA.
Copyright 2006 ACM 1-59593-186-4/06/0007 ...\$5.00.

1. INTRODUCTION

Recent work [1] presented an unbiased comparison between the required computational costs of mutation and crossover on a class of deterministic and stochastic, additively separable problems. Both operators were assumed to possess knowledge of the building blocks (BBs) to operate most efficiently. For noisy problems, the genetic algorithm (GA) using only selection and crossover (a *selectorecombinative* GA) came out on top but the situation flipped for deterministic problems. This dependence on the presence or absence of exogenous noise prompted the question as to how these operators would perform in the presence or absence of fluctuating crosstalk.

Goldberg [2] conjectured that crosstalk or nonlinear interactions of BBs creates similar effects on GA scalability as external noise. This comparison suggested that crosstalk could be handled in the same ways we handle external noise: increase the population size and lengthen the run duration. A validation of this claim was shown in [3] in which fluctuating crosstalk was added to a deterministic fitness function. Required population sizes and convergence times were accurately modeled by known facetwise models of GA scalability in the presence of external noise. Fluctuating crosstalk was modeled using a highest-order Walsh coefficient so that the parity of a solution, represented as a binary chromosome, determined whether a positive or negative constant was added to its fitness already received from the underlying deterministic fitness function. Taking the parity of the entire chromosome represented the case where every building block interacted with every other building block. Results indicated that the noisy scalability models accurately predicted fluctuating crosstalk effects on scalability as long as the strength of the crosstalk was less than the underlying fitness function variance by a certain factor. This factor was empirically observed and a brief intuition behind its derivation was given.

The existence of deterministic noise, and our ability to model its effects on GA scalability, leads us now to consider how sensitive various fitness functions are to varying orders of fluctuating crosstalk. The purpose of this paper is to consider the effects of smaller orders of fluctuating crosstalk on two bounding test cases of problem substructure: uniformly-scaled building blocks and exponentially-scaled building blocks.

The rest of the paper is organized as follows. We first review the literature on epistasis, or nonlinear interactions among bits. We then briefly explain the notion of BBs and depict fluctuating crosstalk in functional form. This view answers the question as to how fluctuating crosstalk effects can be compared to external noise effects. We then present the use of Walsh coefficients to model the crosstalk. Section 4 begins with a review of earlier results showing how fluctuating crosstalk effects on GA scalability can be modeled by facetwise models of exogenous noise. It then continues with uniformly-scaled and exponentially scaled BBs in the presence of fluctuating crosstalk. This paper concludes with a discussion of future work and some implications of our results.

2. LITERATURE REVIEW

Crosstalk, also called inter-BB epistasis, refers to those nonlinear interactions of BBs in a deterministic fitness function. Crosstalk is a specific form of epistasis, or nonlinear interactions of bits or sets of bits. Epistasis has long been known to be a source of difficulty for GAs. Several approaches have tried to either quantitatively measure it, identify it and handle it directly, or to sidestep the problem using another problem representation.

The goal in quantitatively measuring epistasis is to use a metric that directly corresponds to the problem hardness. Davidor [4] reasoned that the amount of epistasis determines problem hardness, but Naudts [5] later observed that the distribution and structure of epistasis also contribute to problem hardness. Davidor’s a priori measurements of problem difficulty and random sampling were shown by Naudts to be sensitive to nonlinear scalings, and Naudts’ own site-wise optimization measure fails to reliably identify problem difficulty. Heckendorn [6] derived local bitwise epistasis measures that correlate well to problem difficulty but only for certain polynomial functions.

Ideally, epistasis could be pinpointed and directly handled. Heckendorn [6] proposes techniques of argument centering and parity truncation to reduce the epistasis for his class of polynomial functions on which his bitwise measures identify. Mühlenbein [7] created a probabilistic model building GA (PMBGA) [8] with multivariate interactions he calls the Factorized Distribution Algorithm. However, this method guarantees optimal solutions for particular gene interactions [9] and requires prior information generally not available.

Finally, sidestepping the epistasis has been sought by changing the problem representation. Both direct approaches [10] and indirect approaches [11] have drawbacks. Direct approaches require clever manipulation, and indirect approaches like Barbulescu’s Gray Code “shifting” technique, require further clarifying of the link between shifting and how it helps reduce to epistasis.

Goldberg [2] proposed an approach that treats crosstalk or inter-BB epistasis like deception, scaling, or external noise depending on the type of the crosstalk. For fluctuating crosstalk, he suggested using facetwise models of population sizing and run duration in the presence of external noise. This approach was validated by [3], upon which this paper extends by considering the effects of varying orders of deterministic noise over various fitness problems. But first, a few preliminaries will be helpful.

3. PRELIMINARIES

Before delving into how fluctuating crosstalk is one facet of problem difficulty and how we model it, we first consider in more detail what GAs are actually processing. We’ve already alluded to the notion of building blocks (BBs) but it would be good to define them more concretely. After that, we discuss how nonlinear interactions of BBs map to problem difficulty.

3.1 GAs Process Minimal Sequentially Superior BBs of a Global Solution

We assume that the GA is optimizing some decomposable or nearly-decomposable fitness function. In other words, there are sets of decision variables that are related to one another. Finding the global solution requires the GA to find which variables are related to another and what values those encoded variables should take on. As the GA processes entire chromosomes, certain bit patterns emerge because certain bits begin to converge as they become more abundant in the most promising solutions found so far. These patterns might change over time depending on the nature of the problem. It is these patterns that we call building blocks because they are substrings of the optimal solution. Since these BBs correspond to the encoded bits of related decision variables, they prove superior to other permutations over the same bit positions in each generation. They are minimal because the competent GA will first find the shorter set of bits common to superior individuals. Over time, longer bit patterns will be found if they contribute more to fitness than any subset did. The goal of the GA then is to find these BBs, and their correct target bits, that when concatenated with bits from other BBs, form an optimal solution. Hence, GAs process what Goldberg [2] calls *minimal sequentially superior BBs of a global solution*, and the interested reader should consult that work for more detail. It is this longer description we refer to when we simply say BBs.

3.2 Crosstalk: Nonlinear Interactions of BBs

Let us suppose that we have a problem as described previously, that is, an additively separable, deterministic, fitness function we call f and define as:

$$f(x) = f_1(x_1x_2x_3) + f_2(x_4x_5) + f_3(x_6x_7)$$

For our discussion here, we use Goldberg’s approach [2] as a foundation. Assume that x is a binary encoding of all decision variables in some specified order and mapping, and that each x_i represents the i^{th} bit of the encoding. This example presupposes that bits x_1 , x_2 , and x_3 are the encoded bits of one or more decision variables that are related to each other and to no other decision variables. The optimal set of bit values to this subfunction, as well as for f_2 and f_3 , must be found to produce a global solution. Hence, this set of bits define one of three BBs. In general, BBs need not have bits directly adjacent to one another, and are not known in advance. However, several Probabilistic Model Building GAs [12] are known to identify BBs and find their target bits values for bounded lengths and structure.

So far, each BB is independent of the other BBs. We now modify f to account for crosstalk, or inter-BB epistasis by introducing a nonlinear BB interaction through f_4 :

$$f(x) = f_1(x_1x_2x_3) + f_2(x_4x_5) + f_3(x_6x_7) + f_4(x_1x_2x_3x_4x_5)$$

f_4 adds some marginal fitness to the overall fitness based on the the first and second BBs. We should be careful to point out that crosstalk doesn't require interactions of entire BBs as in this example but does permit any number of bits from two or more BBs to interact. We also assume that inclusion of f_4 does not change the BB target bits. Goldberg presented a three-way categorization of crosstalk depending on the behavior of the crosstalk function. If f_4 produces a positive value only when bits x_1 to x_5 have reached their target bits, and zero for all other inputs, then it is called *reinforcing crosstalk*. If f_4 produces a negative value only when bits x_1 to x_5 have reached their target bits, and zero for all other inputs, then it is called *punishing crosstalk*. Finally, if f_4 gives a positive value for one parity over any input bits and a negative value for the other parity, it is called *fluctuating crosstalk*. We note that crosstalk can be more general than we describe here if we permit interactions of partial BBs, but the effects on BB processing we will discuss will still hold true. Also, the choice to use parity will become more clear in section 3.4 when we discuss how we principally model fluctuating crosstalk. In that section, it will also be apparent why fluctuating crosstalk can be quite common in fitness functions.

3.3 Crosstalk Maps to Problem Difficulty

Now that we have defined both BBs and crosstalk, we can understand what makes a problem hard for a GA and how each form of crosstalk relates to one facet of problem difficulty. As before, this discussion follows the arguments given by Goldberg [2].

Since the GA processes BBs, intuitively, there are only three sources of problem difficulty: intra-BB difficulty, inter-BB difficulty, and extra-BB difficulty. Goldberg showed that intra-BB difficulty corresponds to deceptive problems where the GA is led to a sub-optimal configuration of bits over the BB. One such example is the trap function, which we use in our experiments and present at a later point. Inter-BB difficulty corresponds to nonuniform scaling of the BBs which requires enough diversity of bits to be present when the GA gets around to solving the least fitness-contributing BBs. We will see an example of this when discussing exponentially scaled BBs. Finally, extra-BB difficulty corresponds to exogenous noise where we treat the noise coming as a single packet and add it to the entire chromosome and not any particular BB.

Perhaps even more interesting than this intuitive division is where crosstalk comes into the picture. Goldberg showed that reinforcing crosstalk can be compared to inter-BB difficulty or scaling, and that punishing crosstalk compares to scaling or deception depending on the strength of the crosstalk. The interested reader may consult the aforementioned work for more detail. We now offer a reasonable explanation of how fluctuating crosstalk maps to extra-BB difficulty or exogenous noise.

It seems unlikely that if a single bit change can bring about such a drastic fitness change (since a parity change results in a sign change of the constant that is added), the GA will converge at all. Yet, if the GA starts with a randomized population, most BBs will not have converged to their target bits in the early generations. Roughly half of the individuals in the population will have even parity and the other half will take on odd parity. Half the individuals will be penalized by some constant w while the other half

will be rewarded by w . This results in a net effect of zero on overall fitness due to the crosstalk. The other key to note is that the population must begin to converge eventually, and once that happens, the fluctuating crosstalk behaves as either reinforcing crosstalk or punishing crosstalk. At this point, we require a GA that can reliably solve problems with punishing or reinforcing crosstalk.

3.4 Principled Modeling of Fluctuating Crosstalk through Walsh Coefficients

The work of Bethke [13] permits us to model any conceivable function, and in our case a fluctuating crosstalk subfunction, over a finite domain of input bits using the Walsh basis. This introduction will be concise and gloss over many details and implications. The goal here is to show how we model fluctuating crosstalk precisely.

For our discussion, we take the intuitive approach introduced by Goldberg [14]. Let $\mathbf{x} = x_l x_{l-1} \dots x_2 x_1$ be the l -bit string representing the coding of the decision variables for an arbitrary individual. We now introduce the auxiliary string positions y_i that are mapped to from the bitwise string positions x_i for $i = 1, \dots, l$ by:

$$y_i = \begin{cases} 1, & \text{if } x_i = 0 \\ -1, & \text{if } x_i = 1. \end{cases}$$

This definition allows the following multiplication to act as an exclusive or operator (XOR). The j^{th} Walsh function $\psi_j(y)$ where $0 \leq j \leq 2^l - 1$ is calculated as:

$$\psi_j(y) = \prod_{i=1}^l y_i^{j_i}, \quad y_i \in \{-1, 1\}$$

where j_i is the i^{th} bit of the binary representation of j and y is understood to be the transformed x using the auxiliary mapping.

For example, for $j = 25 = 11001$ in binary, $\psi_{25}(-1 -1 1 1 -1) = y_1 y_2 y_5 = (-1)(-1)(-1) = -1$. Said another way, $\psi_j(x)$ is a partial-parity function that returns a -1 or a $+1$ as the number of ones in the argument that match the corresponding bits in j is odd or even. Hence, $\psi_{31}(01101) = \psi_{31}(00001) = -1$ since 31 in binary is 11111 and the parity of all bits in the argument are considered.

In the canonical basis, fitness values are obtained by referencing a table of bit strings and their objective fitness values. Bethke [13] showed that any fitness function $f(x)$ over a finite domain can be rewritten as a partial signed sum of the Walsh coefficients, given by

$$f(x) = \sum_{j=0}^{2^l-1} w_j \psi_j(x) .$$

Each Walsh coefficient w_j is real-valued but its exact value is not derived here. Note that each subfunction of the overall fitness function can have multiple, nonzero, Walsh coefficients associated with it. The possible set of Walsh coefficients are dictated by which bit positions correspond to related variables. In our running example, $f(x) = f_1(x_1 x_2 x_3) + f_2(x_4 x_5) + f_3(x_6 x_7) + f_4(x_1 x_2 x_3 x_4 x_5)$, the subfunction f_1 may use any coefficients with index j where $0 \leq j \leq 7$, since this range corresponds to all binary combinations of bits 1 to 3.

For an arbitrary fitness function, we can imagine that some portion of the partial signed sum of the Walsh coeffi-

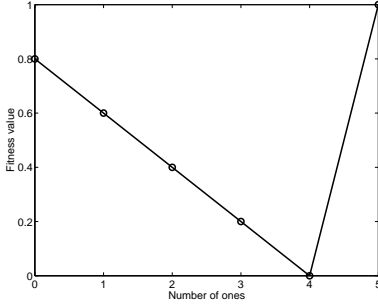


Figure 1: The deceptive 5-bit trap function gives a higher fitness to strings with more zeros than ones unless the optimum at 11111 is found.

cients has direction — meaning that this portion is acyclic in the limit and represents what the GA seeks to solve. The remaining portion fluctuates with possibly irregular periods and shapes - but not contributing to the overall direction of the function. This latter portion will likely involve higher-order Walsh coefficients since a higher order indicates that more bits interact with another. Smaller orders means fewer bits interacting with one another and likely contribute to the direction of the function. Consider the inclusion of the Walsh coefficient w_{2^l-1} for example. This means taking the parity of the entire string. It provides no direction but merely acts as a source of deterministic noise. Other coefficients could be included as part of this fluctuating crosstalk but for this paper we assume only one Walsh coefficient is used. We will vary the order of this coefficient, however, to test GA scalability over different scalings of the underlying fitness function BBs.

4. DETERMINISTIC NOISE EFFECTS ON GA SCALABILITY

Having explained how fluctuating crosstalk maps to exogenous noise and how we principally model it, we now explore its effects on population sizing, run duration, and function evaluations. We start by briefly summarizing our methodology and findings of our earlier work on the subject [3].

4.1 Methodology

Both our current and previous experimentation made use of a *selectorecombinative* GA or a GA using only selection and crossover. Hence, we eliminate any scalability factors due to mutation. We provide a boundedly difficult test function — the concatenated trap function. This function is the sum of multiple subfunctions where each subfunction is a 5-bit trap with a max fitness of 1 at 11111 and other fitness values linearly interpolated from $1 - d$ to 0 as the number of 1's range from 0 to 4. The variable d is called the *signal* and represents the fitness difference between the best and second-best solution. The 5-bit trap is shown in Fig. 1.

The trap misleads the GA since any 5-bit string containing fewer than 4 ones has a higher fitness than another string with more ones, unless the other string is 11111. The tendency of the GA is to increase the frequency of the false peaks at 00000 until the GA finds the higher fitness coming from the infrequent 11111 strings.

The GA utilizes binary tournament selection ($s = 2$) and BB-wise population crossover to avoid disrupting discovered BBs. The use of a BB-respecting operator assumes that the BB structure is known a priori or can be learned, such as through a Probabilistic Model Building GA (PMBGA) [12]. Results for population sizing were obtained by performing 10 to 30 *bisection runs* [15] of 10 to 50 independent GA trials. In a single bisection run, the population size is adjusted after each set of trials until the minimum population size is obtained that yields an average of $m - 1$ correctly discovered BBs. We then report the average of the averages over all bisection runs. Convergence time runs started with a large population size (roughly 10 times the size needed with no noise) and the average of 10 to 50 runs is reported.

4.2 Facetwise Modeling of Deterministic Noise

In this section, we briefly review findings from our previous results that will be helpful to understand current results. For derivations and a more thorough discussion, please refer to [3]. In testing population sizing, we tested required population sizing for functions ranging from 4 to 50 BBs with a signal (the difference in fitness from the best solution to that of the next best solution) ranging from 0.01 to 0.40. Parity was taken over the entire chromosome and a positive or negative constant, a single highest-order Walsh coefficient, was then added to the individual. The magnitude of this constant varied from 0 (no fluctuating crosstalk) to 10 (very high for our problem), mostly in increments of 0.1. An accurate facetwise model of population sizing with exogenous noise, to which we compared our results, comes from Harik, Cantú-Paz, and Goldberg [16] and is given by:

$$n = -\frac{\sqrt{\pi}}{2d} 2^k \log(\alpha) \sqrt{\sigma_f^2 + \sigma_N^2} \quad (1)$$

where d is the *signal*, k is the BB size, α is the error tolerance, σ_f^2 is the fitness function variance, and σ_N^2 is the noise variance. Note that fitness variance $\sigma_f^2 = m\sigma_{BB}^2$ where σ_{BB}^2 is the BB variance and the noise variance $\sigma_N^2 = w_{2^l-1}^2$. With a little work, we derive the ratio of some fitness function with noise to the same function without noise (n_0) to be:

$$n_r = \frac{n}{n_0} = \sqrt{1 + \frac{w_{2^l-1}^2}{m\sigma_{BB}^2}} \quad (2)$$

When the strength of the crosstalk was low, the above model described the effects of the fluctuating crosstalk very well. With very strong crosstalk, half of the initial population was wiped out due to having the wrong parity, and we would expect roughly twice the population needed without noise to be needed with very strong crosstalk. In fact, once that level of crosstalk is exceeded, stronger crosstalk matters little since the concatenated trap still needs to be solved if the optimal solution is to be found, and any more weight to parity doesn't make parity more important since we use tournament selection. If we assume an upper bound of a doubling in required population size, as empirically observed, we can estimate the critical point for which the facetwise model no longer holds as:

$$n_r = 2 = \sqrt{1 + \frac{w_{2^l-1}^2}{m\sigma_{BB}^2}} \quad (3)$$

and hence

$$w_{2^l-1}^2 = 3m\sigma_{BB}^2 \quad (4)$$

The ratio of required time convergence for a function with noise to that without noise is derived from Miller and Goldberg’s [17] time convergence equation:

$$t_c = \frac{\pi\sqrt{l}}{2I} \sqrt{1 + \frac{\sigma_N^2}{\sigma_f^2}}$$

and has the same form as equation 2. We note that we also expect a doubling of run duration since the parity effect may be thought of as a bump in selection pressure since with s -tournament selection the GA now needs $2s$ individuals to have the same *quality of choices* in selection. Individuals of the wrong parity are immediately discarded, and can only be chosen if all s individuals of the wrong parity are placed into the selection pool. This gives the critical divergence from the run duration model to be the same as equation 4. The population sizing ratio and time convergence ratio don’t depend on signal according to our model but empirical results show a slight dependence on the signal.

4.3 Deterministic Noise Effects on Uniformly-Scaled BBs

After reviewing results from the previous work we now consider varying orders of large Walsh coefficients with the same uniform scaling of BBs as before. This answers the question of what effects does fluctuating crosstalk, or deterministic noise, have on GA scalability when those nonlinear interactions of BBs involve fewer BBs but the “noise” coming from them is high. If the strength of the crosstalk is small, we would expect previous facetwise models of GA scalability in the presence of exogenous noise to hold here as well.

In these experiments, there were 50 5-bit traps and parity was taken over the leftmost m' BBs as m' varied from 0 to $m = 50$ in intervals of 2. The magnitude of the corresponding Walsh coefficient was high enough such that if a solution had the wrong parity, the only way it could be selected is if it competed against another solution of the same parity. In this case, a solution of even parity over those m' BBs was rewarded. This meant that the optimal solution of 11111111...11111 was still optimal since the number of bits in the m' BBs was always even.

4.3.1 Time Convergence

Results for the proportion of optimal BBs over time are shown in Fig. 2. The first thing to note is that the bold curve from which the others separate is the plot for $m' = m$ where all BBs are used for parity and that the leftmost plot is for $m' = 0$ or no fluctuating crosstalk. This corroborates our earlier discussion where the noisy problem progresses twice as slowly as when noise is absent and thus requires twice the convergence time. We note that the plot for each m' follows the full-parity plot until some *takeoff* point at which the plot rises sharply and then slows down when convergence is nearly reached. Closer analysis reveals that until that takeoff point, half of the individuals have even parity over those m' BBs and half have odd parity. This explains why all plots with $m' > 0$ follow the $m' = m$ plot for at least some portion: they’re all undergoing the same noise. Yet, even in these noisy conditions, BBs are beginning to converge to 11111. As they do so, each BB tends to lean towards an odd parity but since m' is always even, the net effect is more individuals with even parity over those m' BBs.

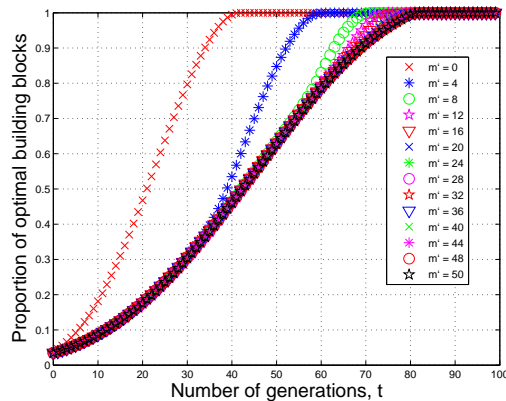


Figure 2: Optimal BB growth for uniformly-scaled BBs as m' varies from 0 to 50. Plots for functions with fluctuating crosstalk follow $m' = m = 50$ plot until parity takeoff. After takeoff, most individuals have correct parity and the GA processes BBs without noise.

We can predict this takeoff point fairly easily if we assume a knowledge of optimal BB growth for the case where $m' = 0$. We justify this assumption since our goal is to investigate the effects of fluctuating crosstalk. We make one more assumption to facilitate the math, and that is that the BB distribution is binomial — meaning that all BBs have converged to 00000 or 11111. This has been found to be a reasonable assumption for $m' \geq 14$ where only 10 % or less of the BBs have not converged to one of those two values. We can estimate what percent of the individuals in the next generation will have even parity over those m' BBs from the current proportion of optimal BBs. This is given by

$$P_{even} = \sum_{k \in E} \binom{m'}{k} p^k (1-p)^{m'-k}$$

where P_{even} is the proportion of solutions with even parity over the m' BBs at time $t + 1$ after BB-wise population crossover, p is the proportion of optimal BBs after selection and reproduction at time t , and E is the set of even numbers from 0 to m' . Since 11111 is odd, and 00000 is even, the only way to produce even parity over m' BBs is to have an even number of 11111 BBs which this equation captures. When P_{even} is just over 0.5, takeoff begins.

At takeoff, the proportion of optimal BBs grows rapidly since as BB convergence leans the population towards even parity, the GA loses less information from the individuals that were discarded by having the wrong parity. And of course, this increase in proportion only results in more even-parity individuals. Within a few generations, this cycle grows the proportion of even individuals to all evens.

This means that we can predict convergence time by finding the time it takes the noiseless plot to reach the p where predicted takeoff occurs and adding to the takeoff time the number of remaining generations to full convergence on the noiseless plot. This simple method results in fairly accurate predictions of time convergence, as seen in Fig. 3

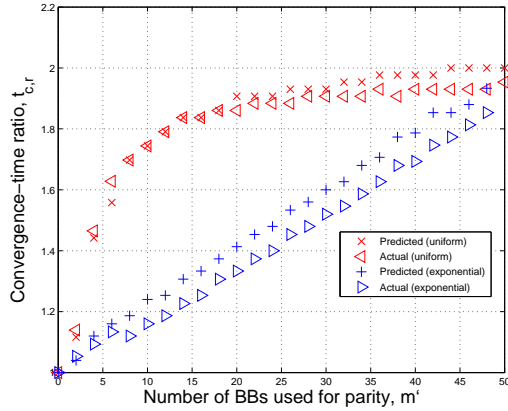


Figure 3: Predicted run durations versus actual run durations. Estimated proportion of even-parity individuals for uniformly-scaled BBs is accurate for $m' \geq 14$.

4.3.2 Population Size

Population sizing can be explained from the simple fact that for each m' , the GA is processing the same magnitude of fluctuating crosstalk where as m (and thus fitness variance) is unchanging. This means that the ratio from equation 2 is constant as m' varies, which as we saw earlier gives a ratio of 2. This means that for each m' , twice the population from that required with no noise is needed. Results are shown in Fig. 4.

4.4 Deterministic Noise Effects on Exponentially-Scaled BBs

This section investigates the effects of fluctuating crosstalk or deterministic noise on exponentially-scaled BBs. The choice to use exponentially-scaled BBs represents the other extreme for BB scaling. A fitness function falling in the middle of these scalings, such as with geometric scaling, will be bounded by the results of these two extremes. Also, if we believe that our fitness function uses one scaling type more than other, we can use the results from the more similar scaling to explain and plan for needed population sizing and run duration.

In an exponentially-scaled deterministic problem with no fluctuating crosstalk, the GA discovers the more salient BBs first since they contribute the most to overall fitness, and the fitness contribution from any given BB is twice the total fitness contribution coming from all BBs of lesser salience. This *sequential* nature of BB convergence has been called *domino convergence* [18].

We should note that in our case, rather than using one BB that is exponentially scaled over the next one, we use groups of four BBs that are exponentially scaled over the next group of four BBs. Within a group of four, each BB is uniformly scaled. This structure ensures that the GA really solves the target bits of the four BBs rather than a potential luck of the draw in having the right configuration of bits for just one BB. Using four BBs also requires a resizing of m to a multiple of four, and in our case, we use $m = 48$. For the following results, the crosstalk is over the most salient m' BBs.

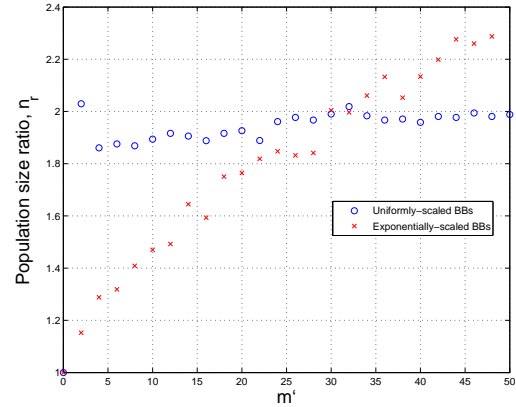


Figure 4: Ratio of population size required for function with fluctuating crosstalk to same function without noise ($m' = 0$). Uniformly-scaled BBs are more sensitive to crosstalk than exponentially-scaled BBs until $m' \geq 30$.

4.4.1 Time Convergence

We start by looking at optimal BB growth as seen in Fig. 5. Results for the exponentially-scaled case are similar to the uniform case at this high level. Once the parity has converged over the m' BBs (shortly after parity takeoff) then the BBs converge at noiseless rates. One critical difference is the BB timing however, which can be seen in Fig. 6 for $m' = 24$.

This graph shows the sequential processing of BBs for $m' = 24$ following the BB processing for $m' = m = 48$ (full-parity) until the parity takeoff; after that, it follows the noiseless model and BBs converge faster. The 12 curves represent the 12 groups of BBs ($m/groupSize = 48/4$) where each curve shows BB growth for all of the four BBs since all four were found to converge at the same rate (since those four were uniformly scaled). Results for $m' \neq 24$ appear similar and are not shown here.

The sequential processing can be explained by the fact that as soon as the target bits for the four BBs in a group are found, in addition to the exponential scaling, the fluctuating crosstalk only reinforces the idea that those individuals should be chosen since m' BBs with bits 11111 will be rewarded for its even parity. The crosstalk is strong enough that individuals with the wrong parity will never be selected unless competing with another individual of the same parity.

The sequential nature suggests a different method to predicting takeoff, namely that we calculate what optimal BB proportion, p , is needed to ensure that an optimal solution over the m' BBs can be found. Since BBs are converging from left to right, this occurs when

$$p^{m'} > \frac{c}{n}$$

where n is the population size and c is some constant number of individuals with an optimal leftmost m' BBs needed for self-sustaining. Empirically, $\frac{c}{n}$ was found to be 0.005. If we assume an understanding of BB growth rates for the noiseless case then we can predict the point of parity takeoff, and also predict convergence times with the same technique

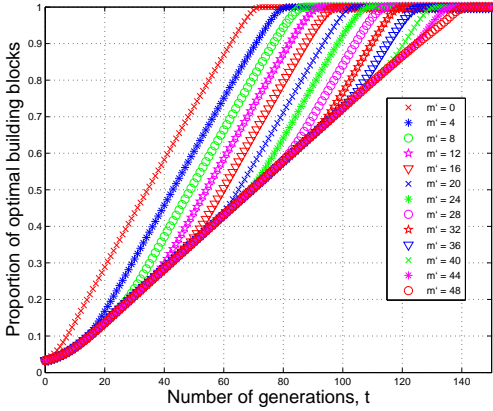


Figure 5: Optimal BB growth for exponentially-scaled BBs as the number of BBs checked for parity, m' , varies from 0 to 50. Plots for functions with fluctuating crosstalk follow $m' = m = 50$ plot until parity takeoff. After takeoff, most individuals have correct parity and the GA processes BBs without noise.

used with uniformly-scaled BBs. Alternatively, we could use our time-convergence model to calculate needed time for 4BBs, and then multiply by the $m'/4$ (the number of groups in m') as a crude estimation. Predicted versus actual run duration ratios are shown in Fig. 3.

4.4.2 Population Size

The sequential nature of the problem means that there must be enough bit diversity remaining in the population to process the last of the m' BBs. It is well known that BBs with very little selection pressure, such as the rightmost BBs, can converge to suboptimal bits due to the accumulation of stochastic errors — a condition known as *genetic drift*. An increase in m' requires a proportionate increase in run duration, which boosts the chances for the rightmost bits to converge due to drift. To overcome this, we must introduce a larger population size to permit more bit diversity from the beginning, so that by the time the GA processes the least salient BBs there is enough bit diversity for the right target bits to be found. Markov chain calculations [19] show that the time to drift t_d is related to population size n by $t_d = 1.4n$. For our case, care must be taken to consider the effects of the crosstalk, and the crucial point to observe is that an increase in population size means a proportionate increase in time before drift occurs. This explains the linear growth of population size in terms of m' that we see in Fig. 4. When $m' = m$, we would expect twice the population size as that needed for no noise as seen from section 4.2.

4.4.3 Function Evaluations

When fitness evaluation time is relatively expensive the number of required fitness evaluations is a good metric for required time. The number of fitness evaluations is the product of time convergence and population size. Although deterministic problems with no fluctuating crosstalk over uniformly-scaled BBs are easier for the GA to solve than are exponentially-scaled BBs, we see from Fig. 7 that uniformly-

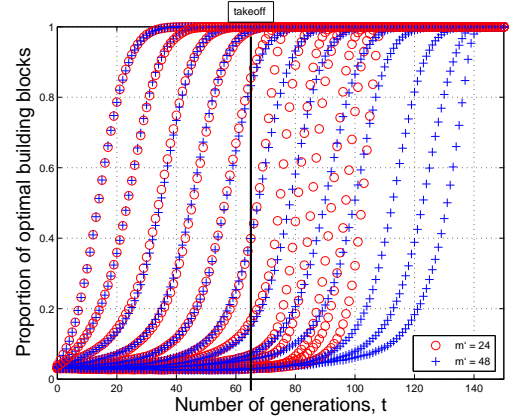


Figure 6: The sequential processing of exponentially-scaled BB groups starting with the most salient 4 BBs (the leftmost curve). BB growth rates for $m' = 24$ follow the full-parity plot ($m' = 48$) until parity takeoff. After takeoff, most individuals have correct parity and the GA processes BBs without noise.

scaled BBs prove more sensitive to fluctuating crosstalk or deterministic noise than do exponentially-scaled BBs. We note that while the function evaluation ratio grows linearly for uniformly-scaled BBs, it grows quadratically (albeit at a very slow rate) for exponentially-scaled BBs since both population size and convergence time grow linearly.

5. FUTURE WORK

Conjectures were made as to fluctuating crosstalk effects on GA scalability for different BB scalings if the strength of the crosstalk were to be reduced. We have yet to validate these claims and hope to do this in the future. Perhaps more interesting though is the question as to how far we can model deterministic noise effects on GA scalability with facetwise models dealing with exogenous noise under varying fitness substructure. Specifically, we plan to consider the introduction of more Walsh coefficients than just one and the use of overlapping BBs. At some point however, we're no longer solving the same problem and BBs that once contributed to a global solution are no longer doing so.

6. SUMMARY & CONCLUSIONS

This study was motivated by a prior performance comparison [1] of mutation to crossover in the presence and absence of exogenous noise. We summarized previous findings of the existence of exogenous noise-like effects on GA scalability in a deterministic fitness function via fluctuating crosstalk. We modeled fluctuating crosstalk with higher-order Walsh coefficients and stated that fluctuating crosstalk behaves like additive exogenous noise until the crosstalk variance far exceeds the underlying fitness variance by a certain threshold we empirically observe. When the crosstalk behaves similarly to external noise, its effects can be handled in a similar manner by increasing the population size and run duration.

We then considered strong fluctuating crosstalk effects on GA scalability for various orders of Walsh coefficients on two extremes of BB scaling — uniformly-scaled and exponentially-

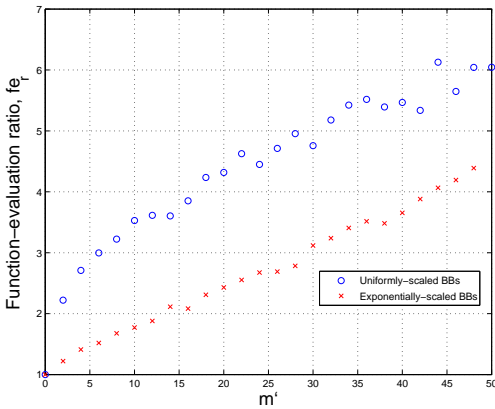


Figure 7: Ratios for required function evaluations of function with fluctuating crosstalk to same function without fluctuating crosstalk.

scaled BBs. Using two extremes of scaling provides bounds as to how an arbitrary fitness function with fluctuating crosstalk will behave. Also, any a priori sense of which scaling is more similar to the crosstalk can serve as a guide to adjusting population sizing and run duration.

7. ACKNOWLEDGMENTS

The authors would like to acknowledge Kumara Sastry for his helpful discussions. This work was sponsored by the Air Force Office of Scientific Research, Air Force Materiel Command, USAF, under grant FA9550-06-1-0096. The U.S. Government is authorized to reproduce and distribute reprints for government purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, or the U.S. Government.

8. REFERENCES

- [1] Sastry, K., Goldberg, D.E.: Let's get ready to rumble: Crossover versus mutation head to head. Proceedings of the 2004 Genetic and Evolutionary Computation Conference **2** (2004) 126–137 Also IlliGAL Report No. 2004005.
- [2] Goldberg, D.E.: Design of Innovation: Lessons from and for Competent Genetic Algorithms. Kluwer Academic Publishers, Boston, MA (2002)
- [3] Sastry, K., Winward, P., Goldberg, D.E., Lima, C.: Fluctuating crosstalk as a source of deterministic noise and its effects on GA scalability. Applications of Evolutionary Computing, EvoWorkshops (2006) 740–751
- [4] Davidor, Y.: Epistasis Variance: A Viewpoint on GA-hardness. foga91 (1991) 23–35
- [5] Naudts, B., Kallel, L.: Some Facts about so-called GA-hardness Measures. Tech. Rep. No. 379, Ecole Polytechnique, CMAP, France (1998)
- [6] Heckendorn, R.B., Whitley, D.: Predicting Epistasis from Mathematical Models. Evolutionary Computation **7**(1) (1999) 69–101
- [7] Mühlenbein, H., Mahnig, T., Rodriguez, A.O.: Schemata, Distributions and Graphical Models in Evolutionary Optimization. Journal of Heuristics **5** (1999) 215–247
- [8] Pelikan, M., Goldberg, D.E., Lobo, F.G.: A Survey of Optimization by Building and Using Probabilistic Models. Comput. Optim. Appl. **21**(1) (2002) 5–20
- [9] Lauritzen, S.L.: Graphical Models. Oxford University Press (1998)
- [10] Beasley, D., Bull, D.R., Martin, R.R.: Reducing Epistasis in Combinatorial Problems by Expansive Coding. In: ICGA. (1993) 400–407
- [11] Barbulescu, L., Watson, J.P., Whitley, L.D.: Dynamic Representations and Escaping Local Optima: Improving Genetic Algorithms and Local Search. In: AAAI/IAAI. (2000) 879–884
- [12] Pelikan, M., Lobo, F., Goldberg, D.E.: A survey of optimization by building and using probabilistic models. Computational Optimization and Applications **21** (2002) 5–20 (Also IlliGAL Report No. 99018).
- [13] Bethke, A.D.: Genetic Algorithms as Function Optimizers. PhD thesis, The University of Michigan (1981)
- [14] Goldberg, D.E.: Genetic Algorithms and Walsh Functions: Part I, a Gentle Introduction. Complex Systems **3**(2) (1989) 129–152 (Also TCGA Report 88006).
- [15] Sastry, K.: Evaluation-Relaxation Schemes for Genetic and Evolutionary Algorithms. Master's thesis, University of Illinois at Urbana-Champaign, General Engineering Department, Urbana, IL (2001) (Also IlliGAL Report No. 2002004).
- [16] Harik, G., Cantú-Paz, E., Goldberg, D.E., Miller, B.L.: The Gambler's Ruin Problem, Genetic Algorithms, and the Sizing of Populations. Evolutionary Computation **7**(3) (1999) 231–253 (Also IlliGAL Report No. 96004).
- [17] Miller, B.L., Goldberg, D.E.: Genetic Algorithms, Selection Schemes, and the Varying Effects of Noise. Evolutionary Computation **4**(2) (1996) 113–131 (Also IlliGAL Report No. 95009).
- [18] Rudnick, M.: Genetic Algorithms and Fitness Variance with an Application to the Automated Design of Artificial Neural Networks. PhD thesis, Oregon Graduate Institute of Science and Technology, Portland (1992)
- [19] Goldberg, D.E., Segrest, P.: Finite Markov chain analysis of genetic algorithms. Proceedings of the Second International Conference on Genetic Algorithms (1987) 1–8